**Expense Tracker:**

import React, { useState } from 'react';

import { View, Text, TextInput, Button, StyleSheet } from 'react-native';

const IncomeExpenseCalculator = () => {

const [income, setIncome] = useState('');

const [expense, setExpense] = useState('');

const [totalExpense, setTotalExpense] = useState(0);

const [remainingIncome, setRemainingIncome] = useState(0);

const addExpense = () => {

if (expense) {

const expenseAmount = parseFloat(expense);

setTotalExpense(totalExpense + expenseAmount);

setExpense('');

}

};

const calculateRemainingIncome = () => {

if (income) {

const incomeAmount = parseFloat(income);

const remaining = incomeAmount - totalExpense;

setRemainingIncome(remaining);

}

};

return (

<View style={styles.container}>

<Text style={styles.title}>Expense Calculator</Text>

<TextInput

style={styles.input}

placeholder="Enter income"

keyboardType="numeric"

value={income}

onChangeText={(text) => setIncome(text)}

/>

<TextInput

style={styles.input}

placeholder="Enter expense"

keyboardType="numeric"

value={expense}

onChangeText={(text) => setExpense(text)}

/>

<Button style={styles.button} title="Add Expense" onPress={addExpense} />

<Button style={styles.button} title="Calculate Remaining Income" onPress={calculateRemainingIncome} />

<Text style={styles.resultText}>

Total Expense: ₹{totalExpense.toFixed(2)}

</Text>

<Text style={styles.resultText}>

Remaining Income: ₹{remainingIncome.toFixed(2)}

</Text>

<Text>715521104001</Text>

</View>

);

};

const styles = StyleSheet.create({

container: {

flex: 1,

padding: 20,

alignItems: 'center',

justifyContent: 'center',

},

title: {

fontSize: 24,

fontWeight: 'bold',

marginBottom: 10,

},

button: {

backgroundColor: 'red',

},

input: {

marginBottom: 10,

borderWidth: 1,

padding: 10,

width: '100%',

fontSize: 20,

},

expenseInputContainer: {

flexDirection: 'row',

alignItems: 'center',

},

resultText: {

fontSize: 20,

fontWeight: 'bold',

marginTop: 20,

},

});

export default IncomeExpenseCalculator;

**Login Form Cordova:**

<!DOCTYPE html>

<html>

<head>

<title>User Login</title>

<link rel="stylesheet" type="text/css" href="css/styles.css">

</head>

<body>

<div class="container">

<h2>Login</h2>

<form id="loginForm">

<label for="username">Username:</label>

<input type="text" id="username" name="username" required>

<br><br>

<label for="password">Password:</label>

<input type="password" id="password" name="password" required>

<br><br>

<button type="button" onclick="resetForm()">Reset</button>

<br><br>

<button type="button" onclick="submitForm()">Submit</button>

<br><br>

</form>

<label id="resultLabel"></label>

</div>

<script type="text/javascript" src="js/index.js"></script>

</body>

</html>

JS:

function resetForm() {

document.getElementById('loginForm').reset();

document.getElementById('resultLabel').innerHTML = '';

}

function submitForm() {

var username = document.getElementById('username').value;

var password = document.getElementById('password').value;

// Add your login logic here

if (username === 'example' && password === 'password') {

document.getElementById('resultLabel').innerHTML = 'Login successful!';

} else {

document.getElementById('resultLabel').innerHTML = 'Invalid username or password. Please try again.';

}

}

**Geolocation Cordova:**

</html>

<!DOCTYPE html>

<html>

<head>

<meta charset="utf-8" />

<meta name="viewport" content="width=device-width, initial-scale=1">

<title>Location App</title>

</head>

<body>

<h1>Location App</h1>

<p id="location"></p>

<script type="text/javascript" src="js/index.js"></script>

</body>

</html>

JS:

document.addEventListener('deviceready', onDeviceReady, false);

function onDeviceReady() {

navigator.geolocation.getCurrentPosition(onSuccess, onError);

}

function onSuccess(position) {

var element = document.getElementById('location');

element.innerHTML = 'Latitude: ' + position.coords.latitude + '<br />' +

'Longitude: ' + position.coords.longitude;

}

function onError(error) {

alert('Error getting location: ' + error.message);

}

**BMI APP(Native):**

import React,{useState} from 'react';

import {View,Text,Button,TextInput,StyleSheet} from 'react-native';

const bmiapp=()=>{

const [weight,setWeight]=useState("");

const [height,setHeight]=useState("");

const[bmi,setBmi]=useState(0);

const calcBmi=()=>{

const w=parseFloat(weight);

const h=parseFloat(height);

const a=w/h\*\*2;

setBmi(a);

};

return(

<View style={styles.container}>

<Text style={styles.text}>BMI CALCULATOR</Text>

<TextInput

style={styles.input}

placeholder="enter weight in kg"

value={weight}

onChangeText={(text)=>setWeight(text)}

/>

<TextInput

style={styles.input}

placeholder="enter height in m"

value={height}

onChangeText={(text)=>setHeight(text)}

/>

<Button title="calculate" onPress={calcBmi}/>

<Text style={styles.text}>The BMI is {bmi.toFixed(2)}</Text>

</View>

)

}

const styles=StyleSheet.create({

container:{

flex:1,

padding:20,

},

text:{

size:18

},

input:{

width:'100%',

size:18

}

});

export default bmiapp;

**COLORCHANGE(Native):**

import React,{useState} from 'react';

import {Button,Text,View} from 'react-native';

const colorChange=()=>{

const [color,setColor]=useState('');

const textColor=()=>{

setColor("#FF0000");

};

return(

<View>

<Text style={{color:color}}>Hello World</Text>

<Button title="Click Me" onPress={textColor}/>

</View> )}export default colorChange;

**IMPERIAL CONVERTER(NATIVE):**

import React, { useState } from 'react';

import { Button, StyleSheet, Text, TextInput, View } from 'react-native';

const UnitConverterApp = () => {

const [km, setKm] = useState('');

const [miles, setMiles] = useState('');

const [kg, setKg] = useState('');

const [pounds, setPounds] = useState('');

const convertKmToMiles = () => {

const milesValue = parseFloat(km) \* 0.621371;

setMiles(milesValue.toFixed(2));

};

const convertKgToPounds = () => {

const poundsValue = parseFloat(kg) \* 2.20462;

setPounds(poundsValue.toFixed(2));

};

return (

<View style={styles.container}>

<Text style={styles.title}>Unit Converter</Text>

<View style={styles.converterContainer}>

<Text style={styles.label}>Kilometers to Miles</Text>

<TextInput

style={styles.input}

placeholder="Enter kilometers"

keyboardType="numeric"

value={km}

onChangeText={(text) => setKm(text)}

/>

<Button title="Convert" onPress={convertKmToMiles} />

<Text style={styles.result}>{miles} miles</Text>

</View>

<View style={styles.converterContainer}>

<Text style={styles.label}>Kilograms to Pounds</Text>

<TextInput

style={styles.input}

placeholder="Enter kilograms"

keyboardType="numeric"

value={kg}

onChangeText={(text) => setKg(text)}

/>

<Button title="Convert" onPress={convertKgToPounds} />

<Text style={styles.result}>{pounds} pounds</Text>

</View>

</View>

);

};

const styles = StyleSheet.create({

container: {

flex: 1,

justifyContent: 'center',

alignItems: 'center',

},

title: {

fontSize: 24,

fontWeight: 'bold',

marginBottom: 20,

},

converterContainer: {

marginBottom: 20,

width: '80%',

},

label: {

fontSize: 18,

marginBottom: 10,

},

input: {

height: 40,

borderColor: 'gray',

borderWidth: 1,

paddingHorizontal: 10,

marginBottom: 10,

},

result: {

marginTop: 10,

fontSize: 16,

fontWeight: 'bold',

},

});

export default UnitConverterApp;

**TEXT ALIGN(Native):**

import React,{useState} from 'react';

import {View,Text,Button,StyleSheet} from 'react-native';

const textAlign=()=>{

const [textalign,settextalign]=useState('center');

return(

<View>

<Text style={{textAlign:textalign}}>HEY THERE!</Text>

<Button title="Left" onPress={()=>settextalign("Left")}/>

<Button title="Right" onPress={()=>settextalign("Right")}/>

<Button title="Center" onPress={()=>settextalign("Center")}/>

</View>

)

};

export default textAlign;

**LOGIN PAGE(Native) :**

import React,{useState} from 'react';

import {View,Text,TextInput,Button,StyleSheet,Alert} from 'react-native';

const loginapp=()=>{

const [username,setUserName]=useState("");

const [password,setPassword]=useState("");

const [msg,setMsg]=useState("");

const verify=()=>{

if(username==='user' && password==="password"){

Alert.alert("Login Successful");

setMsg("Success");

}

else{

Alert.alert("Invalid Entry");

setMsg("Try Again");

}

};

return(

<View style={styles.container}>

<Text style={styles.text}>Enter Username:</Text>

<TextInput style={styles.input} onChangeText={(text)=>setUserName(text)} value={username}/>

<Text style={styles.text}>Enter Password:</Text>

<TextInput style={styles.input} onChangeText={(text)=>setPassword(text)} value={password}/>

<Button title="login" onPress={verify}/>

<Text>{msg}</Text>

</View>

)

}

const styles=StyleSheet.create({

container:{

alignItems:'center',

justifyContents:'center'

},

text:{

size:20

},

input:{

size:20,

borderWidth:1,

borderColor:'gray'

}

});

export default loginapp;

**TEXT COLOR(ANDROID):**

import android.graphics.Color;

import android.os.Bundle;

import android.support.v7.app.AppCompatActivity; import android.view.View;

import android.widget.Button;

import android.widget.TextView;

public class MainActivity extends AppCompatActivity { private TextView textView;

@Override protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState); setContentView(R.layout.activity\_main);

// Find the TextView by its ID textView = findViewById(R.id.textView);

// Find the Button by its ID

Button changeColorButton = findViewById(R.id.changeColorButton);

// Set an OnClickListener for the button changeColorButton.setOnClickListener(new View.OnClickListener() {

@Override public void onClick(View v) { // Change the text color to red textView.setTextColor(Color.RED);

}

});

}]}

//xml

<!-- res/layout/activity\_main.xml -->

<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android" xmlns:tools="http://schemas.android.com/tools" android:layout\_width="match\_parent" android:layout\_height="match\_parent" android:paddingLeft="16dp" android:paddingTop="16dp" android:paddingRight="16dp" android:paddingBottom="16dp" tools:context=".MainActivity">

<TextView android:id="@+id/textView" android:layout\_width="wrap\_content" android:layout\_height="wrap\_content" android:text="Hello, World!" android:textSize="20sp" android:layout\_centerHorizontal="true" android:layout\_marginBottom="20dp"/>

<Button android:id="@+id/changeColorButton" android:layout\_width="wrap\_content" android:layout\_height="wrap\_content" android:text="Change Color" android:layout\_below="@id/textView" android:layout\_centerHorizontal="true"/>

</RelativeLayout>

**ALERT MESSAGE(ANDROID):**

//java

import android.os.Bundle;

import android.support.v7.app.AppCompatActivity;

import android.view.View;

import android.widget.Button;

import android.widget.Toast;

public class MainActivity extends AppCompatActivity {

@Override protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

Button showAlertButton = findViewById(R.id.showAlertButton);

showAlertButton.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

// Display a short toast message

Toast.makeText(MainActivity.this, "Alert: This is a text alert!", Toast.LENGTH\_SHORT).show();

}

});

}

}

//xml

<Button android:id="@+id/showAlertButton" android:layout\_width="wrap\_content" android:layout\_height="wrap\_content" android:text="Show Alert" android:layout\_below="@id/textView" android:layout\_centerHorizontal="true"/>

**FORM(ANDROID):**

//java

import android.os.Bundle;

import android.support.v7.app.AppCompatActivity; import android.view.View; import android.widget.Button; import android.widget.EditText; import android.widget.Toast; public class MainActivity extends AppCompatActivity {

@Override protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState); setContentView(R.layout.activity\_main);

final EditText editTextName = findViewById(R.id.editTextName); final EditText editTextEmail = findViewById(R.id.editTextEmail); Button showInfoButton = findViewById(R.id.showInfoButton);

showInfoButton.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

String name = editTextName.getText().toString();

String email = editTextEmail.getText().toString();

String message = "Name: " + name + "\nEmail: " + email;

// Display a Toast message with entered information

Toast.makeText(MainActivity.this, message, Toast.LENGTH\_SHORT).show();

}

});

}

}

//xml

<!-- res/layout/activity\_main.xml -->

<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android" xmlns:tools="http://schemas.android.com/tools" android:layout\_width="match\_parent" android:layout\_height="match\_parent" android:orientation="vertical" android:padding="16dp" tools:context=".MainActivity">

<EditText android:id="@+id/editTextName" android:layout\_width="match\_parent" android:layout\_height="wrap\_content" android:hint="Enter your name"/>

<EditText android:id="@+id/editTextEmail" android:layout\_width="match\_parent" android:layout\_height="wrap\_content" android:inputType="textEmailAddress" android:hint="Enter your email"/>

<Button android:id="@+id/showInfoButton" android:layout\_width="wrap\_content" android:layout\_height="wrap\_content" android:text="Show Info"/>

</LinearLayout>

**LIBRARY MANAGEMENT:**

git clone <https://github.com/codehunter-sk/Lib_Mgt_02.git>

**TODO LIST:**

import 'package:flutter/material.dart';

void main() {

runApp(TodoApp());

}

class TodoApp extends StatelessWidget {

@override

Widget build(BuildContext context) {

return MaterialApp(

title: 'Todo App',

theme: ThemeData(

primarySwatch: Colors.blue,

),

home: TodoScreen(),

);

}

}

class TodoScreen extends StatefulWidget {

@override

\_TodoScreenState createState() => \_TodoScreenState();

}

class \_TodoScreenState extends State<TodoScreen> {

final List<String> \_tasks = [];

final TextEditingController \_taskController = TextEditingController();

void \_addTask() {

setState(() {

final task = \_taskController.text;

if (task.isNotEmpty) {

\_tasks.add(task);

\_taskController.clear();

}

});

}

void \_removeTask(int index) {

setState(() {

\_tasks.removeAt(index);

});

}

@override

Widget build(BuildContext context) {

return Scaffold(

appBar: AppBar(

title: Text('Todo List Application'),

),

body: Column(

children: <Widget>[

Padding(

padding: const EdgeInsets.all(20.0),

child: TextField(

controller: \_taskController,

decoration: InputDecoration(

hintText: 'Enter new task',

),

),

),

ElevatedButton(

onPressed: \_addTask,

child: Text('Add Task'),

),

Expanded(

child: ListView.builder(

itemCount: \_tasks.length,

itemBuilder: (context, index) {

return ListTile(

title: Text(\_tasks[index]),

trailing: IconButton(

icon: Icon(Icons.delete),

onPressed: () => \_removeTask(index),

),

);

},

),

),

Text(

'715521104004',

),

],

),

);

}

}