**Flutter Lab File**

**Chirag Bargoojar**

**02611104421**

**Practical Set 1**

**1) Write a Dart program to print the Fibonacci series without using recursion and using recursion.**

**Ans.**

Without Recursion

void main() {

void printFibonacciSeries(int n) {

int a = 0, b = 1, c;

print(a);

print(b);

for (int i = 2; i < n; i++) {

c = a + b;

print(c);

a = b;

b = c;

}

}

printFibonacciSeries(5);

}



With Recursion

void main() {

void printFibonacciSeries(int n, int a, int b) {

if (n == 0) {

return;

}

print(a);

printFibonacciSeries(n - 1, b, a + b);

}

printFibonacciSeries(10, 0, 1);

}



**2) Write a Dart program to check a prime number.**

**Ans.**

**import 'dart:math';**

**void main() {**

**bool isPrime(int n) {**

**if (n <= 1) {**

**return false;**

**}**

**for (int i = 2; i <= sqrt(n); i++) {**

**if (n % i == 0) {**

**return false;**

**}**

**}**

**return true;**

**}**

**int num = 11;**

**print(isPrime(num) ? "$num is PRIME" : "$num is NOT PRIME");**

**}**

****

**3) Write a Dart program to check the palindrome number.**

**Ans.**

**void main() {**

**bool isPalindrome(String s) {**

**for (int i = 0, j = s.length - 1; i < j; i++, j--) {**

**if (s[i] != s[j]) {**

**return false;**

**}**

**}**

**return true;**

**}**

**String str = "racecar";**

**print(isPalindrome(str) ? "$str is PALINDROME" : "$str is NOT PALINDROME");**

**}**

****

**4) Write a Dart program to print the factorial of a number.**

**Ans.**

**void main() {**

**int fact(int num) {**

**int result = 1;**

**for (int i = 1; i <= num; i++) {**

**result \*= i;**

**}**

**return result;**

**}**

**int num = 7;**

**print("${fact(num)} is factorial of $num");**

**}**

****

**5) Write a Dart program to print the sum of digits.**

**Ans.**

**void main() {**

**int sumOfDigits(int n) {**

**int sum = 0;**

**while (n > 0) {**

**sum += n % 10;**

**n = n ~/ 10;**

**}**

**return sum;**

**}**

**int num = 12345;**

**print("${sumOfDigits(num)} is sum of digit of $num");**

**}**

****

**6) Make a list of factors of a given number.**

**Ans.**

**void main() {**

**List<int> factors(int n) {**

**List<int> factors = [];**

**for (int i = 1; i <= n; i++) {**

**if (n % i == 0) {**

**factors.add(i);**

**}**

**}**

**return factors;**

**}**

**int num = 20;**

**print("${factors(num)} is list of factors of $num");**

**}**

****

**7) Make a list of all prime factors of a given number.**

**Ans.**

**import 'dart:math';**

**void main() {**

**List<int> primeFactors(int n) {**

**List<int> factors = [];**

**while (n % 2 == 0) {**

**factors.add(2);**

**n = n ~/ 2;**

**}**

**for (int i = 3; i <= sqrt(n); i += 2) {**

**while (n % i == 0) {**

**factors.add(i);**

**n = n ~/ i;**

**}**

**}**

**if (n > 2) {**

**factors.add(n);**

**}**

**return factors;**

**}**

**int num = 20;**

**print("${primeFactors(num)} is list of prime factors of $num");**

**}**

****

**8) Finding the Number of Factors of a given number.**

**Ans.**

**import 'dart:math';**

**void main() {**

**int numberOfFactors(int n) {**

**int count = 0;**

**for (int i = 1; i <= sqrt(n); i++) {**

**if (n % i == 0) {**

**count++;**

**if (n / i != i) {**

**count++;**

**}**

**}**

**}**

**return count;**

**}**

**int num = 20;**

**print("${numberOfFactors(num)} is number of factors of $num");**

**}**

****

**9) Write a program to reverse a number.**

**Ans.**

**void main() {**

**int number = 12345;**

**String reversedNumber =**

**int.parse(number.toString().split('').reversed.join()).toString();**

**print("Reverse of $number is $reversedNumber");**

**}**

****

**10) Write a program to print the number into words Input N=125 Output: One Two Five .**

**Ans.**

**void main() {**

**Map<int, String> digitMap = {**

**0: "Zero",**

**1: "One",**

**2: "Two",**

**3: "Three",**

**4: "Four",**

**5: "Five",**

**6: "Six",**

**7: "Seven",**

**8: "Eight",**

**9: "Nine",**

**};**

**int num = 125;**

**List<String> numSplit = "$num".split("");**

**String word = numSplit.map((e) => digitMap[int.parse(e)]).toList().join(" ");**

**print(word);**

**}**

****

**11) Write a program to print the number into words Input N=125 Output: One hundred twenty five.**

**Ans.**

**void main() {**

**String convertToWords(int n) {**

**List<String> belowTwenty = [**

**'',**

**'One',**

**'Two',**

**'Three',**

**'Four',**

**'Five',**

**'Six',**

**'Seven',**

**'Eight',**

**'Nine',**

**'Ten',**

**'Eleven',**

**'Twelve',**

**'Thirteen',**

**'Fourteen',**

**'Fifteen',**

**'Sixteen',**

**'Seventeen',**

**'Eighteen',**

**'Nineteen'**

**];**

**List<String> belowHundred = [**

**'',**

**'Ten',**

**'Twenty',**

**'Thirty',**

**'Forty',**

**'Fifty',**

**'Sixty',**

**'Seventy',**

**'Eighty',**

**'Ninety'**

**];**

**if (n < 20) {**

**return belowTwenty[n];**

**} else if (n < 100) {**

**return belowHundred[n ~/ 10] +**

**(n % 10 != 0 ? ' ' + belowTwenty[n % 10] : '');**

**} else if (n < 1000) {**

**return belowTwenty[n ~/ 100] + ' Hundred ' + convertToWords(n % 100);**

**}**

**return '';**

**}**

**int n = 999;**

**print(convertToWords(n));**

**}**

****

**12) Write a program to print binary equivalent of a number Input: 8 Output: 100.**

**Ans.**

**void main() {**

**String convertToBinary(int n) {**

**String binary = "";**

**while (n > 0) {**

**int bit = n & 1;**

**binary = "$bit" + binary;**

**n = n >> 1;**

**}**

**return binary;**

**}**

**int n = 100;**

**print(convertToBinary(n));**

**}**

****

**13) Write a program to create a room class, the attributes of this class is roomno, roomtype, roomarea and ACmachine. In this class the member functions are setdata and displaydata. Write the main function to declare the objects of the class and call the member functions.**

**Ans.**

**class Room {**

**late int roomNo;**

**late String roomType;**

**late double roomArea;**

**late bool hasAcMachine;**

**void setData(int no, String type, double area, bool ac) {**

**roomNo = no;**

**roomType = type;**

**roomArea = area;**

**hasAcMachine = ac;**

**}**

**void displayData() {**

**print("Room Number: $roomNo");**

**print("Room Type: $roomType");**

**print("Room Area: $roomArea sq.ft");**

**print("AC Machine: $hasAcMachine");**

**}**

**}**

**void main() {**

**Room room1 = Room();**

**Room room2 = Room();**

**room1.setData(101, "Single", 200.0, true);**

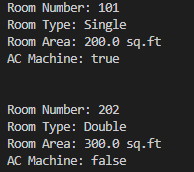
**room2.setData(202, "Double", 300.0, false);**

**room1.displayData();**

**print("\n");**

**room2.displayData();**

**}**

****

**Practical Set 2**

**Q1. Write a program to create the Application to perform the following operations on the number Increment, Decrement, Square, Reset Check Prime, Check Even/Odd, Check Palindrome, Compute Factorial.**

**Ans.**

**import 'dart:math';**

**import 'package:flutter/material.dart';**

**void main() {**

**runApp(const MyApp());**

**}**

**class MyApp extends StatelessWidget {**

**const MyApp({super.key});**

**@override**

**Widget build(BuildContext context) {**

**return MaterialApp(**

**title: 'Flutter Demo',**

**debugShowCheckedModeBanner: false,**

**theme: ThemeData(**

**primarySwatch: Colors.blue,**

**),**

**home: const MyHomePage(),**

**);**

**}**

**}**

**class MyHomePage extends StatefulWidget {**

**const MyHomePage({super.key});**

**@override**

**State<MyHomePage> createState() => \_MyHomePageState();**

**}**

**class \_MyHomePageState extends State<MyHomePage> {**

**int counter = 0;**

**String operationString = "";**

**String operationString2 = "";**

**void inc() {**

**setState(() {**

**counter++;**

**});**

**}**

**void dec() {**

**setState(() {**

**counter--;**

**});**

**}**

**void square() {**

**setState(() {**

**operationString = "Your have tried to check the SQUARE of the counter";**

**operationString2 = "Square of your number is ${counter \* counter}";**

**});**

**}**

**void reset() {**

**setState(() {**

**counter = 0;**

**operationString = "";**

**operationString2 = "";**

**});**

**}**

**void prime() {**

**bool isPrime(int number) {**

**if (number <= 1) {**

**return false;**

**}**

**int limit = sqrt(number).floor();**

**for (int i = 2; i <= limit; i++) {**

**if (number % i == 0) {**

**return false;**

**}**

**}**

**return true;**

**}**

**setState(() {**

**operationString = "Your have tried to check the PRIME of the counter";**

**operationString2 =**

**isPrime(counter) ? "$counter is PRIME" : "$counter is not PRIME";**

**});**

**}**

**void evenOdd() {**

**setState(() {**

**operationString = "Your have tried to check the Even/Odd of the counter";**

**operationString2 =**

**counter % 2 == 0 ? "$counter is EVEN" : "$counter is ODD";**

**});**

**}**

**void palindrome() {**

**bool isPalindrome(int num) {**

**String original = num.toString();**

**String reversed = original.split('').reversed.join();**

**return original == reversed;**

**}**

**setState(() {**

**operationString =**

**"Your have tried to check the PALINDROME of the counter";**

**operationString2 = isPalindrome(counter)**

**? "$counter is PALINDROME"**

**: "$counter is NOT PALINDROME";**

**});**

**}**

**void factorial() {**

**int fact(int num) {**

**int result = 1;**

**for (int i = 1; i <= num; i++) {**

**result \*= i;**

**}**

**return result;**

**}**

**setState(() {**

**operationString =**

**"Your have tried to check the PALINDROME of the counter";**

**operationString2 = "Factorial of $counter is ${fact(counter)}";**

**});**

**}**

**Widget myText(String text) {**

**return FittedBox(**

**child: Text(text),**

**);**

**}**

**@override**

**Widget build(BuildContext context) {**

**return Scaffold(**

**appBar: AppBar(**

**title: Text("Flutter First App"),**

**),**

**body: SizedBox(**

**child: Center(**

**child: Column(**

**crossAxisAlignment: CrossAxisAlignment.center,**

**mainAxisAlignment: MainAxisAlignment.center,**

**children: [**

**Text("Your have pushed the button this many times:"),**

**Text(**

**counter.toString(),**

**style: TextStyle(**

**fontSize: 25,**

**),**

**),**

**Text(operationString),**

**Text(**

**operationString2,**

**style: TextStyle(**

**fontSize: 22,**

**),**

**),**

**Padding(**

**padding: const EdgeInsets.all(8.0),**

**child: Column(**

**children: [**

**Row(**

**mainAxisAlignment: MainAxisAlignment.center,**

**children: [**

**FloatingActionButton(**

**onPressed: inc,**

**child: myText("+"),**

**),**

**FloatingActionButton(**

**onPressed: dec,**

**child: myText("-"),**

**),**

**FloatingActionButton(**

**onPressed: square,**

**child: myText("Square"),**

**),**

**FloatingActionButton(**

**onPressed: reset,**

**child: myText("Reset"),**

**),**

**],**

**),**

**Row(**

**mainAxisAlignment: MainAxisAlignment.center,**

**children: [**

**FloatingActionButton(**

**onPressed: prime,**

**child: myText("Prime"),**

**),**

**FloatingActionButton(**

**onPressed: evenOdd,**

**child: myText("Even/Odd"),**

**),**

**FloatingActionButton(**

**onPressed: palindrome,**

**child: myText("Palindrome"),**

**),**

**FloatingActionButton(**

**onPressed: factorial,**

**child: myText("Factorial"),**

**),**

**],**

**),**

**],**

**),**

**),**

**],**

**),**

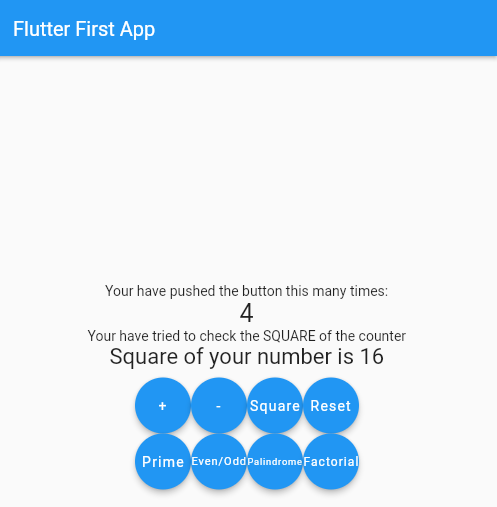
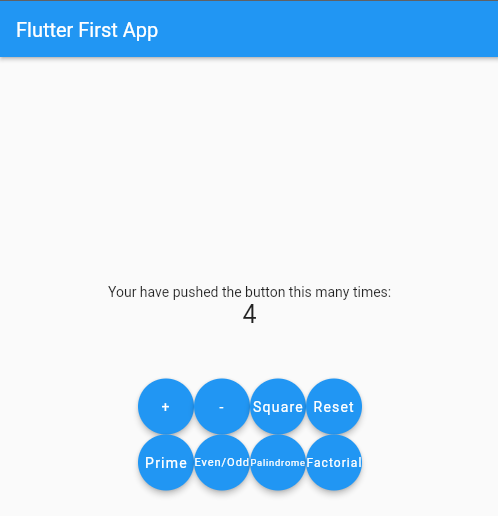
**),**

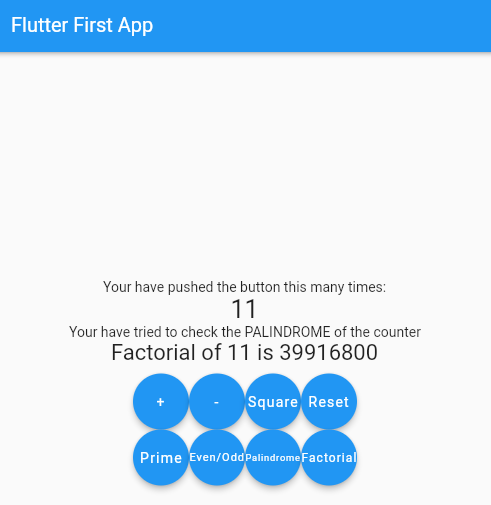
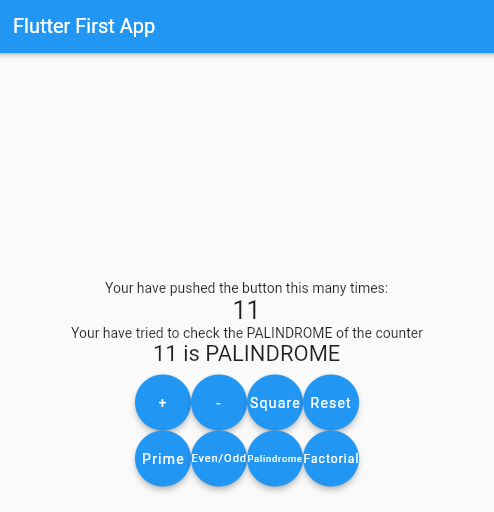
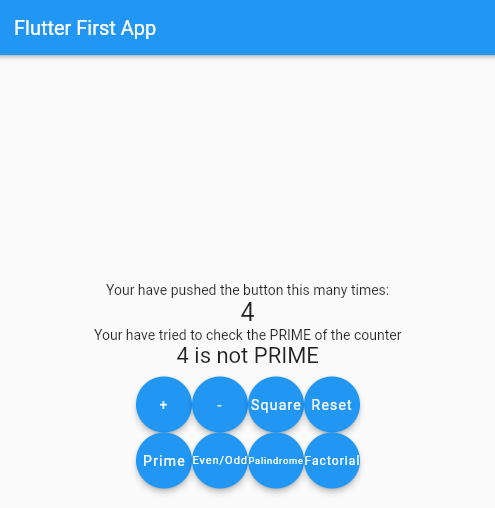
**),**

**);**

**}**

**}**

****

****

**Q2. Create an application using Bottom Navigation Bar, Bottom app Bar and tabbar.**

**Ans.**

**import 'package:flutter/material.dart';**

**void main() {**

**runApp(const MyApp());**

**}**

**class MyApp extends StatelessWidget {**

**const MyApp({super.key});**

**@override**

**Widget build(BuildContext context) {**

**return MaterialApp(**

**title: 'Flutter Demo',**

**debugShowCheckedModeBanner: false,**

**theme: ThemeData(**

**primarySwatch: Colors.blue,**

**),**

**home: const MyHomePage(),**

**);**

**}**

**}**

**class MyHomePage extends StatefulWidget {**

**const MyHomePage({super.key});**

**@override**

**State<MyHomePage> createState() => \_MyHomePageState();**

**}**

**class \_MyHomePageState extends State<MyHomePage> {**

**@override**

**Widget build(BuildContext context) {**

**List<IconData> list = [**

**Icons.call,Icons.camera,Icons.chat,Icons.phone,Icons.category,**

**];**

**List<String> list2 = [**

**"Calls","Camera","Chats",”Phone","List",**

**];**

**return DefaultTabController(**

**initialIndex: 0,**

**length: 4,**

**child: Scaffold(**

**floatingActionButtonLocation: FloatingActionButtonLocation.centerDocked,**

**floatingActionButton: FloatingActionButton(**

**onPressed: () {},**

**child: Icon(Icons.add),**

**),**

**bottomNavigationBar: BottomAppBar(**

**color: Colors.blue[900],**

**shape: CircularNotchedRectangle(),**

**notchMargin: 5,**

**child: Row(**

**mainAxisSize: MainAxisSize.max,**

**mainAxisAlignment: MainAxisAlignment.spaceBetween,**

**children: <Widget>[**

**IconButton(**

**icon: Icon(**

**Icons.menu,**

**color: Colors.white,**

**),**

**onPressed: () {},**

**),**

**Spacer(),**

**IconButton(**

**icon: Icon(**

**Icons.search,**

**color: Colors.white,**

**),**

**onPressed: () {},**

**),**

**IconButton(**

**icon: Icon(**

**Icons.more\_vert\_rounded,**

**color: Colors.white,**

**),**

**onPressed: () {},**

**),**

**],**

**),**

**),**

**appBar: AppBar(**

**title: const Text('BottomNavigationBar Demo'),**

**bottom: const TabBar(**

**tabs: <Widget>[**

**Tab(**

**text: "Incoming Call",**

**icon: null,**

**),**

**Tab(**

**text: "Outgoing Call",**

**icon: null,**

**),**

**Tab(**

**text: "Missed Call",**

**icon: null,**

**),**

**Tab(**

**text: "Contacts",**

**icon: null,**

**),**

**],**

**),**

**),**

**body: Column(**

**children: [**

**Expanded(**

**child: TabBarView(**

**children: <Widget>[**

**Offstage(),**

**Offstage(),**

**Offstage(),**

**Offstage(),**

**],**

**),**

**),**

**SizedBox(**

**height: 70,**

**child: Row(**

**mainAxisAlignment: MainAxisAlignment.spaceBetween,**

**children: [**

**for (int i = 0; i < list.length; i++)**

**Expanded(**

**child: Column(**

**children: [**

**Icon(**

**list[i],**

**),**

**Text(list2[i]),**

**],**

**),**

**),**

**],**

**),**

**),**

**],**

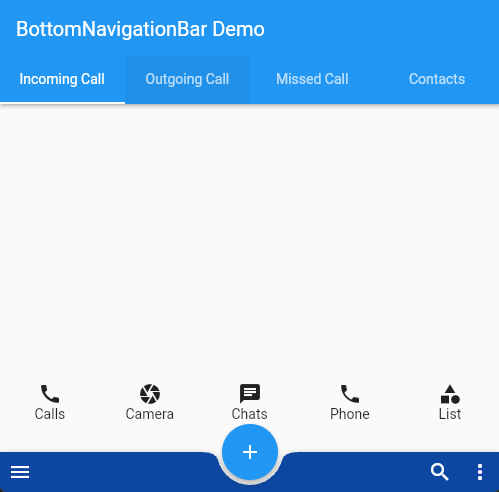
**),**

**),**

**);**

**}**

**}**

****

**Q3. Create a Flutter application that obtains two integers from the user and prints their product, difference and quotient (division).**

**Ans.**

**import 'package:flutter/material.dart';**

**void main() {**

**runApp(const MyApp());**

**}**

**class MyApp extends StatelessWidget {**

**const MyApp({super.key});**

**@override**

**Widget build(BuildContext context) {**

**return MaterialApp(**

**title: 'Flutter Demo',**

**debugShowCheckedModeBanner: false,**

**theme: ThemeData(**

**primarySwatch: Colors.blue,**

**),**

**home: const MyHomePage(),**

**);**

**}**

**}**

**class MyHomePage extends StatefulWidget {**

**const MyHomePage({super.key});**

**@override**

**State<MyHomePage> createState() => \_MyHomePageState();**

**}**

**class \_MyHomePageState extends State<MyHomePage> {**

**var ct1 = TextEditingController();**

**var ct2 = TextEditingController();**

**String difference = "";**

**String product = "";**

**String quotient = "";**

**void compute() {**

**try {**

**setState(() {**

**difference =**

**"Difference = ${int.parse(ct1.text) - int.parse(ct2.text)}";**

**product = "Product = ${int.parse(ct1.text) \* int.parse(ct2.text)}";**

**quotient = "Quotient = ${int.parse(ct1.text) / int.parse(ct2.text)}";**

**});**

**} catch (e) {**

**print("Error computing $e");**

**}**

**}**

**Widget myText(String text) {**

**return Text(**

**text,**

**style: TextStyle(**

**fontSize: 21,**

**),**

**);**

**}**

**@override**

**Widget build(BuildContext context) {**

**return Scaffold(**

**appBar: AppBar(**

**title: Text("App"),**

**),**

**body: Padding(**

**padding: const EdgeInsets.all(8.0),**

**child: SizedBox(**

**child: Column(**

**mainAxisAlignment: MainAxisAlignment.center,**

**crossAxisAlignment: CrossAxisAlignment.center,**

**children: [**

**TextField(**

**controller: ct1,**

**decoration: const InputDecoration(**

**labelText: "First Number",**

**border: OutlineInputBorder(),**

**),**

**),**

**Padding(**

**padding: const EdgeInsets.only(top: 8.0),**

**child: TextField(**

**controller: ct2,**

**decoration: const InputDecoration(**

**labelText: "Second Number",**

**border: OutlineInputBorder(),**

**),**

**),**

**),**

**myText(difference),**

**myText(product),**

**myText(quotient),**

**ElevatedButton(**

**onPressed: compute,**

**child: Text(**

**"COMPUTE",**

**style: TextStyle(**

**fontSize: 21,**

**),**

**),**

**)**

**],**

**),**

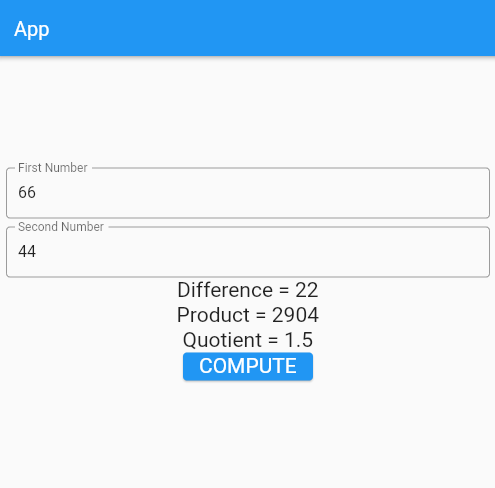
**),**

**),**

**);**

**}**

**}**

****

**Q4. Create a flutter Application to perform the following operations on the number Converts a Fahrenheit to degree Celsius using the formula: Celsius = [(Fahrenheit Value)- 32] \* 5 / 9 Converts a degree Celsius to Fahrenheit using the formula: Fahrenheit=[(Celsius value) × (9/5)] + 32.**

**Ans.**

**import 'package:flutter/material.dart';**

**void main() {**

**runApp(const MyApp());**

**}**

**class MyApp extends StatelessWidget {**

**const MyApp({super.key});**

**@override**

**Widget build(BuildContext context) {**

**return MaterialApp(**

**title: 'Flutter Demo',**

**debugShowCheckedModeBanner: false,**

**theme: ThemeData(**

**primarySwatch: Colors.blue,**

**),**

**home: const MyHomePage(),**

**);**

**}**

**}**

**class MyHomePage extends StatefulWidget {**

**const MyHomePage({super.key});**

**@override**

**State<MyHomePage> createState() => \_MyHomePageState();**

**}**

**class \_MyHomePageState extends State<MyHomePage> {**

**var ct1 = TextEditingController();**

**String celsicus = "";**

**String farh = "";**

**void toCelcius() {**

**try {**

**setState(() {**

**celsicus = "Celsicus = ${(int.parse(ct1.text) - 32) \* 5 / 9}";**

**farh = '';**

**});**

**} catch (e) {**

**print("Error computing $e");**

**}**

**}**

**void toFarhn() {**

**try {**

**setState(() {**

**farh = "Fahrenheit = ${(int.parse(ct1.text) \* (9 / 5)) + 32}";**

**celsicus = '';**

**});**

**} catch (e) {**

**print("Error computing $e");**

**}**

**}**

**Widget myText(String text) {**

**return Text(**

**text,**

**style: TextStyle(**

**fontSize: 21,**

**),**

**);**

**}**

**@override**

**Widget build(BuildContext context) {**

**return Scaffold(**

**appBar: AppBar(**

**title: Text("App"),**

**),**

**body: Padding(**

**padding: const EdgeInsets.all(8.0),**

**child: SizedBox(**

**child: Column(**

**mainAxisAlignment: MainAxisAlignment.center,**

**crossAxisAlignment: CrossAxisAlignment.center,**

**children: [**

**TextField(**

**controller: ct1,**

**decoration: const InputDecoration(**

**labelText: "Enter Number",**

**border: OutlineInputBorder(),**

**),**

**),**

**myText(celsicus),**

**myText(farh),**

**ElevatedButton(**

**onPressed: toCelcius,**

**child: Text(**

**"Computer to Celcius",**

**style: TextStyle(**

**fontSize: 21,**

**),**

**),**

**),**

**SizedBox(**

**height: 10,**

**),**

**ElevatedButton(**

**onPressed: toFarhn,**

**child: Text(**

**"Computer to Fahrenheit",**

**style: TextStyle(**

**fontSize: 21,**

**),**

**),**

**),**

**],**

**),**

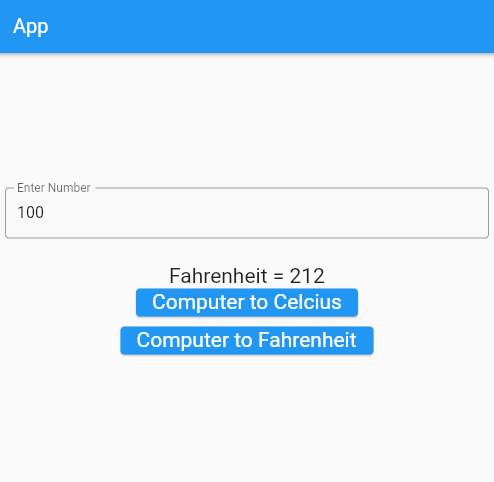
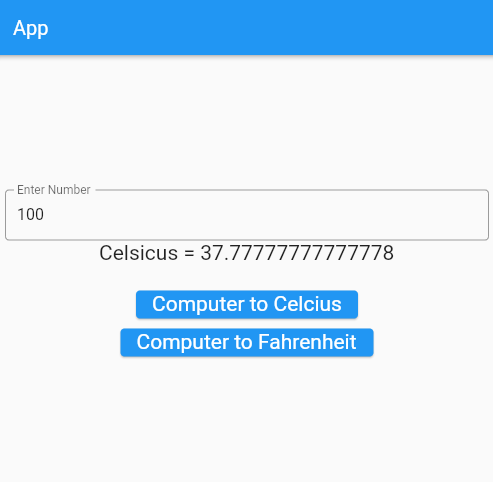
**),**

**),**

**);**

**}**

**}**

****

**Q5. Create a Flutter application with the following UI to display the image of the image url of TextField.**

**Ans.**

**import 'package:flutter/material.dart';**

**void main() {**

**runApp(const MyApp());**

**}**

**class MyApp extends StatelessWidget {**

**const MyApp({super.key});**

**@override**

**Widget build(BuildContext context) {**

**return MaterialApp(**

**title: 'Flutter Demo',**

**debugShowCheckedModeBanner: false,**

**theme: ThemeData(**

**primarySwatch: Colors.blue,**

**),**

**home: const MyHomePage(),**

**);**

**}**

**}**

**class MyHomePage extends StatefulWidget {**

**const MyHomePage({super.key});**

**@override**

**State<MyHomePage> createState() => \_MyHomePageState();**

**}**

**class \_MyHomePageState extends State<MyHomePage> {**

**var ct1 = TextEditingController();**

**String imageLink = "";**

**@override**

**Widget build(BuildContext context) {**

**ct1.addListener(() {**

**setState(() {**

**imageLink = ct1.text;**

**});**

**});**

**return Scaffold(**

**appBar: AppBar(**

**title: Text("App"),**

**),**

**body: Column(**

**children: [**

**Padding(**

**padding: const EdgeInsets.all(8.0),**

**child: TextField(**

**controller: ct1,**

**decoration: const InputDecoration(**

**labelText: "Full Image Path",**

**border: OutlineInputBorder(),**

**),**

**),**

**),**

**SizedBox(**

**height: 10,**

**),**

**Image.network(**

**imageLink,**

**errorBuilder: (context, error, stackTrace) {**

**return const Text("Image path invalid");**

**},**

**),**

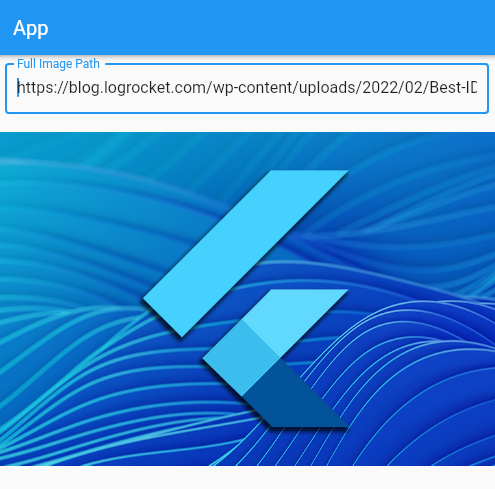
**],**

**),**

**);**

**}**

**}**

****

**Q6. Create an application using Inkwell , OnTap event widgets.**

**Ans.**

**import 'package:flutter/material.dart';**

**void main() {**

**runApp(const MyApp());**

**}**

**class MyApp extends StatelessWidget {**

**const MyApp({super.key});**

**@override**

**Widget build(BuildContext context) {**

**return MaterialApp(**

**title: 'Flutter Demo',**

**debugShowCheckedModeBanner: false,**

**theme: ThemeData(**

**primarySwatch: Colors.blue,**

**),**

**home: const MyHomePage(),**

**);**

**}**

**}**

**class MyHomePage extends StatefulWidget {**

**const MyHomePage({super.key});**

**@override**

**State<MyHomePage> createState() => \_MyHomePageState();**

**}**

**class \_MyHomePageState extends State<MyHomePage> {**

**@override**

**Widget build(BuildContext context) {**

**return Scaffold(**

**appBar: AppBar(**

**title: Text("App"),**

**),**

**body: Center(**

**child: Column(**

**crossAxisAlignment: CrossAxisAlignment.center,**

**mainAxisAlignment: MainAxisAlignment.center,**

**children: [**

**SizedBox(**

**height: 100,**

**width: 100,**

**child: InkWell(**

**onTap: () {**

**print("Card 1 tapped");**

**},**

**child: Card(**

**elevation: 10,**

**child: Center(child: Text("Card 1")),**

**),**

**),**

**),**

**SizedBox(**

**height: 100,**

**width: 100,**

**child: InkWell(**

**onTap: () {**

**print("Card 2 tapped");**

**},**

**child: Card(**

**elevation: 10,**

**child: Center(child: Text("Card 2")),**

**),**

**),**

**),**

**],**

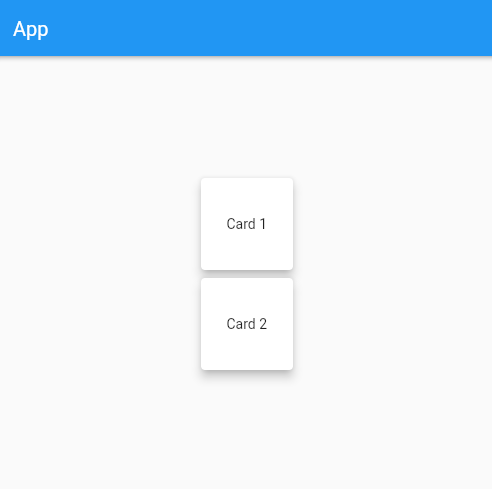
**),**

**),**

**);**

**}**

**}**

****

**Q7. Create an application using Draggable, Drag target widgets**

**Ans.**

**import 'package:flutter/material.dart';**

**void main() {**

**runApp(const MyApp());**

**}**

**class MyApp extends StatelessWidget {**

**const MyApp({super.key});**

**@override**

**Widget build(BuildContext context) {**

**return MaterialApp(**

**title: 'Flutter Demo',**

**debugShowCheckedModeBanner: false,**

**theme: ThemeData(**

**primarySwatch: Colors.blue,**

**),**

**home: const MyHomePage(),**

**);**

**}**

**}**

**class MyHomePage extends StatefulWidget {**

**const MyHomePage({super.key});**

**@override**

**State<MyHomePage> createState() => \_MyHomePageState();**

**}**

**class \_MyHomePageState extends State<MyHomePage> {**

**bool \_isDropped = false;**

**String \_color = "done";**

**String imageLink = "";**

**@override**

**Widget build(BuildContext context) {**

**return Scaffold(**

**body: Container(**

**child: Center(**

**child: Column(**

**mainAxisAlignment: MainAxisAlignment.center,**

**children: [**

**Draggable<String>(**

**// Data is the value this Draggable stores.**

**data: \_color,**

**child: Container(**

**height: 150.0,**

**width: 150.0,**

**color: Colors.redAccent,**

**child: Center(**

**child: Image.asset(**

**"assets/images/mickey1.png",**

**),**

**),**

**),**

**feedback: Material(**

**child: Container(**

**height: 170.0,**

**width: 170.0,**

**decoration: BoxDecoration(**

**color: Colors.redAccent,**

**),**

**child: Center(**

**child: Image.asset(**

**"assets/images/mickey1.png",**

**),**

**),**

**),**

**),**

**childWhenDragging: Container(**

**height: 150.0,**

**width: 150.0,**

**color: Colors.grey,**

**child: Center(**

**child: Image.asset(**

**"assets/images/mickey1.png",**

**),**

**),**

**),**

**),**

**SizedBox(**

**height: MediaQuery.of(context).size.height \* 0.15,**

**),**

**DragTarget<String>(**

**builder: (**

**BuildContext context,**

**List<dynamic> accepted,**

**List<dynamic> rejected,**

**) {**

**return SizedBox(**

**child: ClipRRect(**

**borderRadius: BorderRadius.all(Radius.circular(12)),**

**child: Container(**

**height: 200,**

**width: 200,**

**color: \_isDropped ? Colors.redAccent : null,**

**child: !\_isDropped**

**? Center(**

**child: Image.asset(**

**"assets/images/mickey2.png",**

**),**

**)**

**: Center(**

**child: Image.asset(**

**"assets/images/mickey1.png",**

**),**

**)**

**),**

**),**

**);**

**},**

**onAccept: (data) {**

**debugPrint('hi $data');**

**setState(() {**

**\_isDropped = true;**

**});**

**},**

**onWillAccept: (data) {**

**return data == \_color;**

**},**

**onLeave: (data) {},**

**),**

**],**

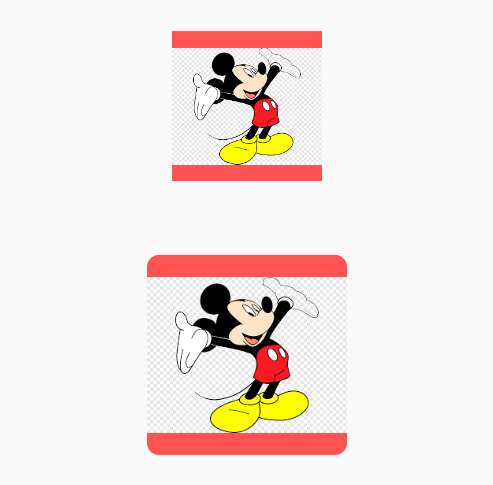
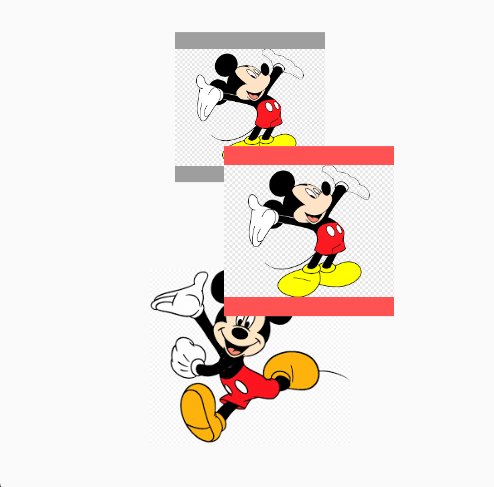
**),**

**),**

**));**

**}**

**}**

****

**Q8. Create the Tic Tac Toe App**

**Ans.**

**import 'package:flutter/material.dart';**

**void main() {**

**runApp(const MyApp());**

**}**

**class MyApp extends StatelessWidget {**

**const MyApp({super.key});**

**@override**

**Widget build(BuildContext context) {**

**return MaterialApp(**

**title: 'Flutter Demo',**

**debugShowCheckedModeBanner: false,**

**theme: ThemeData(**

**primarySwatch: Colors.blue,**

**),**

**home: HomePage(),**

**);**

**}**

**}**

**class HomePage extends StatefulWidget {**

**@override**

**\_HomePageState createState() => \_HomePageState();**

**}**

**class \_HomePageState extends State<HomePage> {**

**bool oTurn = true;**

**// 1st player is O**

**List<String> displayElement = ['', '', '', '', '', '', '', '', ''];**

**int oScore = 0;**

**int xScore = 0;**

**int filledBoxes = 0;**

**@override**

**Widget build(BuildContext context) {**

**return Scaffold(**

**backgroundColor: Colors.indigo[900],**

**body: Column(**

**children: <Widget>[**

**Expanded(**

**child: Container(**

**child: Row(**

**mainAxisAlignment: MainAxisAlignment.center,**

**children: <Widget>[**

**Padding(**

**padding: const EdgeInsets.all(30.0),**

**child: Column(**

**mainAxisAlignment: MainAxisAlignment.center,**

**children: <Widget>[**

**Text(**

**'Player X',**

**style: TextStyle(**

**fontSize: 20,**

**fontWeight: FontWeight.bold,**

**color: Colors.white),**

**),**

**Text(**

**xScore.toString(),**

**style: TextStyle(fontSize: 20, color: Colors.white),**

**),**

**],**

**),**

**),**

**Padding(**

**padding: const EdgeInsets.all(30.0),**

**child: Column(**

**mainAxisAlignment: MainAxisAlignment.center,**

**children: <Widget>[**

**Text('Player O',**

**style: TextStyle(**

**fontSize: 20,**

**fontWeight: FontWeight.bold,**

**color: Colors.white)),**

**Text(**

**oScore.toString(),**

**style: TextStyle(fontSize: 20, color: Colors.white),**

**),**

**],**

**),**

**),**

**],**

**),**

**),**

**),**

**Expanded(**

**flex: 4,**

**child: GridView.builder(**

**itemCount: 9,**

**gridDelegate: SliverGridDelegateWithFixedCrossAxisCount(**

**crossAxisCount: 3),**

**itemBuilder: (BuildContext context, int index) {**

**return GestureDetector(**

**onTap: () {**

**\_tapped(index);**

**},**

**child: Container(**

**decoration: BoxDecoration(**

**border: Border.all(color: Colors.white)),**

**child: Center(**

**child: Text(**

**displayElement[index],**

**style: TextStyle(color: Colors.white, fontSize: 35),**

**),**

**),**

**),**

**);**

**}),**

**),**

**Expanded(**

**child: Container(**

**child: Row(**

**mainAxisAlignment: MainAxisAlignment.center,**

**children: <Widget>[**

**ElevatedButton(**

**onPressed: \_clearScoreBoard,**

**child: Text("Clear Score Board"),**

**),**

**],**

**),**

**))**

**],**

**),**

**);**

**}**

**void \_tapped(int index) {**

**setState(() {**

**if (oTurn && displayElement[index] == '') {**

**displayElement[index] = 'O';**

**filledBoxes++;**

**} else if (!oTurn && displayElement[index] == '') {**

**displayElement[index] = 'X';**

**filledBoxes++;**

**}**

**oTurn = !oTurn;**

**\_checkWinner();**

**});**

**}**

**void \_checkWinner() {**

**// Checking rows**

**if (displayElement[0] == displayElement[1] &&**

**displayElement[0] == displayElement[2] &&**

**displayElement[0] != '') {**

**\_showWinDialog(displayElement[0]);**

**}**

**if (displayElement[3] == displayElement[4] &&**

**displayElement[3] == displayElement[5] &&**

**displayElement[3] != '') {**

**\_showWinDialog(displayElement[3]);**

**}**

**if (displayElement[6] == displayElement[7] &&**

**displayElement[6] == displayElement[8] &&**

**displayElement[6] != '') {**

**\_showWinDialog(displayElement[6]);**

**}**

**// Checking Column**

**if (displayElement[0] == displayElement[3] &&**

**displayElement[0] == displayElement[6] &&**

**displayElement[0] != '') {**

**\_showWinDialog(displayElement[0]);**

**}**

**if (displayElement[1] == displayElement[4] &&**

**displayElement[1] == displayElement[7] &&**

**displayElement[1] != '') {**

**\_showWinDialog(displayElement[1]);**

**}**

**if (displayElement[2] == displayElement[5] &&**

**displayElement[2] == displayElement[8] &&**

**displayElement[2] != '') {**

**\_showWinDialog(displayElement[2]);**

**}**

**// Checking Diagonal**

**if (displayElement[0] == displayElement[4] &&**

**displayElement[0] == displayElement[8] &&**

**displayElement[0] != '') {**

**\_showWinDialog(displayElement[0]);**

**}**

**if (displayElement[2] == displayElement[4] &&**

**displayElement[2] == displayElement[6] &&**

**displayElement[2] != '') {**

**\_showWinDialog(displayElement[2]);**

**} else if (filledBoxes == 9) {**

**\_showDrawDialog();**

**}**

**}**

**void \_showWinDialog(String winner) {**

**showDialog(**

**barrierDismissible: false,**

**context: context,**

**builder: (BuildContext context) {**

**return AlertDialog(**

**title: Text("\" " + winner + " \" is Winner!!!"),**

**actions: [**

**TextButton(**

**child: Text("Play Again"),**

**onPressed: () {**

**\_clearBoard();**

**Navigator.of(context).pop();**

**},**

**)**

**],**

**);**

**});**

**if (winner == 'O') {**

**oScore++;**

**} else if (winner == 'X') {**

**xScore++;**

**}**

**}**

**void \_showDrawDialog() {**

**showDialog(**

**barrierDismissible: false,**

**context: context,**

**builder: (BuildContext context) {**

**return AlertDialog(**

**title: Text("Draw"),**

**actions: [**

**TextButton(**

**child: Text("Play Again"),**

**onPressed: () {**

**\_clearBoard();**

**Navigator.of(context).pop();**

**},**

**)**

**],**

**);**

**});**

**}**

**void \_clearBoard() {**

**setState(() {**

**for (int i = 0; i < 9; i++) {**

**displayElement[i] = '';**

**}**

**});**

**filledBoxes = 0;**

**}**

**void \_clearScoreBoard() {**

**setState(() {**

**xScore = 0;**

**oScore = 0;**

**for (int i = 0; i < 9; i++) {**

**displayElement[i] = '';**

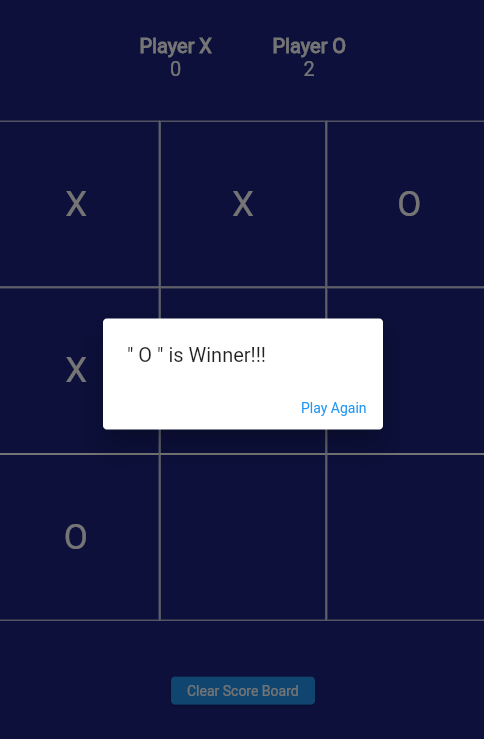
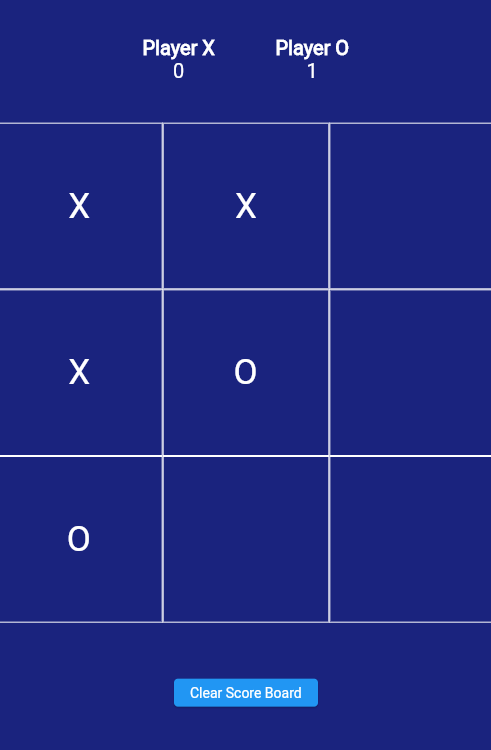
**}**

**});**

**filledBoxes = 0;**

**}**

**}**

****

**Q9. Create the app to display Toast Message.**

**Ans.**

**import 'package:flutter/material.dart';**

**void main() {**

**runApp(const MyApp());**

**}**

**class MyApp extends StatelessWidget {**

**const MyApp({super.key});**

**@override**

**Widget build(BuildContext context) {**

**return MaterialApp(**

**title: 'Flutter Demo',**

**debugShowCheckedModeBanner: false,**

**theme: ThemeData(**

**primarySwatch: Colors.blue,**

**),**

**home: const MyHomePage(),**

**);**

**}**

**}**

**class MyHomePage extends StatefulWidget {**

**const MyHomePage({super.key});**

**@override**

**State<MyHomePage> createState() => \_MyHomePageState();**

**}**

**class \_MyHomePageState extends State<MyHomePage> {**

**var ct1 = TextEditingController();**

**@override**

**Widget build(BuildContext context) {**

**void showToast() {**

**ScaffoldMessenger.of(context).showSnackBar(**

**SnackBar(**

**behavior: SnackBarBehavior.floating,**

**duration: const Duration(seconds: 10),**

**content: Text(**

**"Your message: ${ct1.text}",**

**),**

**),**

**);**

**}**

**return Scaffold(**

**appBar: AppBar(**

**title: Text("App"),**

**),**

**body: Column(**

**children: [**

**Padding(**

**padding: const EdgeInsets.all(8.0),**

**child: TextField(**

**controller: ct1,**

**decoration: const InputDecoration(**

**labelText: "Enter Message",**

**border: OutlineInputBorder(),**

**),**

**),**

**),**

**ElevatedButton(**

**onPressed: showToast,**

**child: Text(**

**"Display Toast",**

**),**

**),**

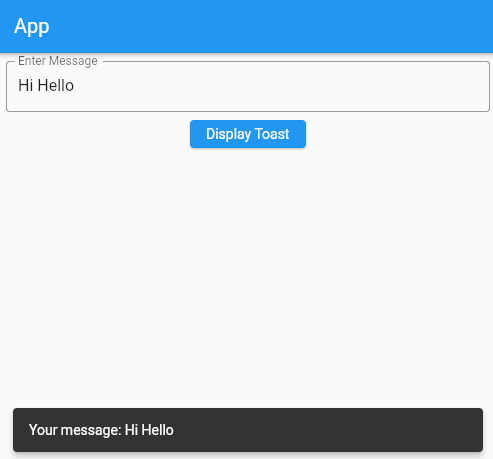
**],**

**),**

**);**

**}**

**}**

****

**Q10. Design a login Page and navigate to the next page if the correct credentials are entered.**

**Ans.**

**import 'package:flutter/material.dart';**

**void main() {**

**runApp(const MyApp());**

**}**

**class MyApp extends StatelessWidget {**

**const MyApp({super.key});**

**@override**

**Widget build(BuildContext context) {**

**return MaterialApp(**

**title: 'Flutter Demo',**

**debugShowCheckedModeBanner: false,**

**theme: ThemeData(**

**primarySwatch: Colors.blue,**

**),**

**home: const MyHomePage(),**

**);**

**}**

**}**

**class MyHomePage extends StatefulWidget {**

**const MyHomePage({super.key});**

**@override**

**State<MyHomePage> createState() => \_MyHomePageState();**

**}**

**class \_MyHomePageState extends State<MyHomePage> {**

**var ct1 = TextEditingController();**

**var ct2 = TextEditingController();**

**void navigateTo() {**

**String user = "admin";**

**String password = "admin";**

**if (ct1.text.trim() == user && ct2.text.trim() == password) {**

**Navigator.pushReplacement(**

**context,**

**MaterialPageRoute(**

**builder: (context) => LoginPage(),**

**),**

**);**

**} else {**

**ScaffoldMessenger.of(context).showSnackBar(**

**const SnackBar(**

**behavior: SnackBarBehavior.floating,**

**duration: const Duration(seconds: 10),**

**content: Text(**

**"Invalid Login Credentials",**

**),**

**),**

**);**

**}**

**}**

**@override**

**Widget build(BuildContext context) {**

**return Scaffold(**

**appBar: AppBar(**

**title: Text("App"),**

**),**

**body: Padding(**

**padding: const EdgeInsets.all(8.0),**

**child: SizedBox(**

**child: Column(**

**mainAxisAlignment: MainAxisAlignment.center,**

**crossAxisAlignment: CrossAxisAlignment.center,**

**children: [**

**Text(**

**"LOGIN",**

**style: TextStyle(**

**fontSize: 25,**

**color: Colors.blue,**

**),**

**),**

**TextField(**

**controller: ct1,**

**decoration: const InputDecoration(**

**labelText: "Username",**

**border: OutlineInputBorder(),**

**),**

**),**

**Padding(**

**padding: const EdgeInsets.only(top: 8.0),**

**child: TextField(**

**controller: ct2,**

**obscureText: true,**

**decoration: const InputDecoration(**

**labelText: "Password",**

**border: OutlineInputBorder(),**

**),**

**),**

**),**

**SizedBox(**

**width: double.infinity,**

**child: Padding(**

**padding: const EdgeInsets.all(8.0),**

**child: ElevatedButton(**

**onPressed: navigateTo,**

**child: Text(**

**"Login",**

**style: TextStyle(**

**fontSize: 21,**

**),**

**),**

**),**

**),**

**)**

**],**

**),**

**),**

**),**

**);**

**}**

**}**

**class LoginPage extends StatelessWidget {**

**const LoginPage({super.key});**

**@override**

**Widget build(BuildContext context) {**

**return Scaffold(**

**appBar: AppBar(**

**title: Text("Home Page"),**

**),**

**body: Center(**

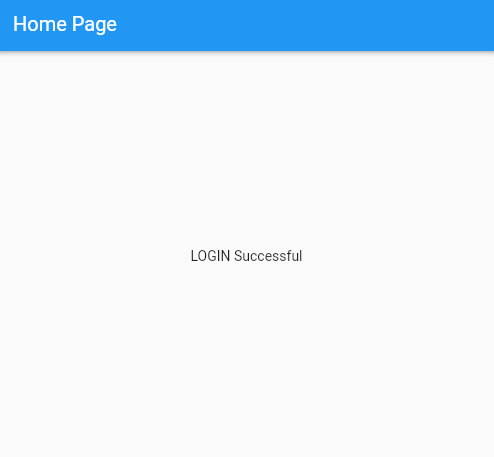
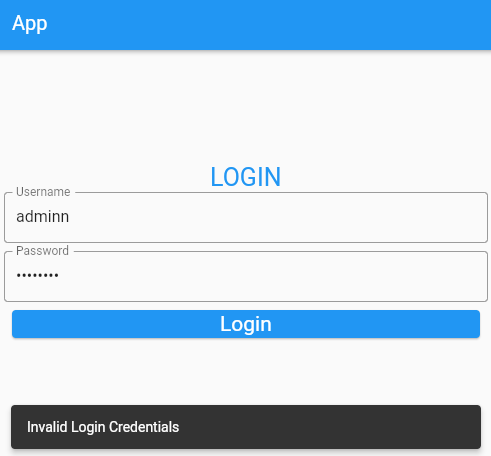
**child: Text("LOGIN Successful"),**

**),**

**);**

**}**

**}**

****

**Q11. Create the application to open the stated URL on the click of the image**

**Ans.**

* Add the package to pubspec.yaml

url\_launcher: ^6.1.7

import 'package:flutter/material.dart';

import 'package:url\_launcher/url\_launcher.dart';

void main() {

runApp(const MyApp());

}

class MyApp extends StatelessWidget {

const MyApp({super.key});

@override

Widget build(BuildContext context) {

return MaterialApp(

title: 'Flutter Demo',

debugShowCheckedModeBanner: false,

theme: ThemeData(

primarySwatch: Colors.blue,

),

home: const MyHomePage(),

);

}

}

class MyHomePage extends StatefulWidget {

const MyHomePage({super.key});

@override

State<MyHomePage> createState() => \_MyHomePageState();

}

class \_MyHomePageState extends State<MyHomePage> {

@override

Widget build(BuildContext context) {

return Scaffold(

appBar: AppBar(

title: Text("App"),

),

body: Center(

child: InkWell(

onTap: () {

String \_url = "https://flutter.dev";

Future<void> \_launchUrl() async {

await launchUrl(

Uri.parse(\_url),

webOnlyWindowName: '\_blank',

);

}

\_launchUrl();

},

child: FlutterLogo(

size: 100,

),

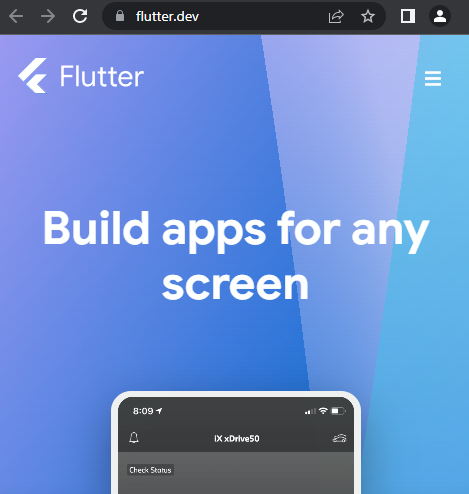
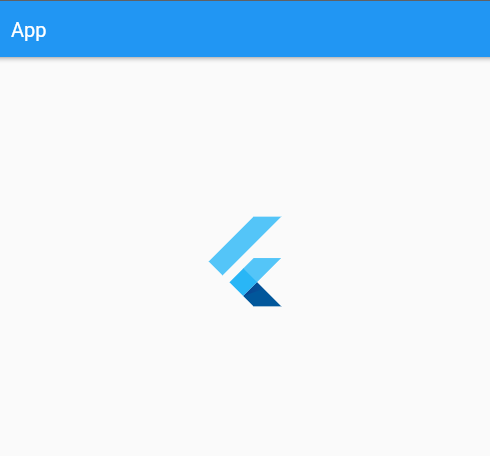
),

),

);

}

}



**Q12. Create an app to play and pause a Video**

**Ans.**

* Add the package to pubspec.yaml

video\_player: ^2.4.10

**//////////////////////////////////////////////////////////////////////////**

**import 'package:flutter/material.dart';**

**import 'package:video\_player/video\_player.dart';**

**void main() {**

**runApp(const MyApp());**

**}**

**class MyApp extends StatelessWidget {**

**const MyApp({super.key});**

**@override**

**Widget build(BuildContext context) {**

**return MaterialApp(**

**title: 'Flutter Demo',**

**debugShowCheckedModeBanner: false,**

**theme: ThemeData(**

**primarySwatch: Colors.blue,**

**),**

**home: const MyHomePage(),**

**);**

**}**

**}**

**class MyHomePage extends StatefulWidget {**

**const MyHomePage({super.key});**

**@override**

**State<MyHomePage> createState() => \_MyHomePageState();**

**}**

**class \_MyHomePageState extends State<MyHomePage> {**

**late VideoPlayerController \_controller;**

**@override**

**void initState() {**

**super.initState();**

**\_controller = VideoPlayerController.network(**

**'https://flutter.github.io/assets-for-api-docs/assets/videos/bee.mp4')**

**..initialize().then((\_) {**

**setState(() {});**

**});**

**}**

**@override**

**Widget build(BuildContext context) {**

**return Scaffold(**

**appBar: AppBar(**

**title: Text("Video App"),**

**),**

**body: Center(**

**child: \_controller.value.isInitialized**

**? AspectRatio(**

**aspectRatio: \_controller.value.aspectRatio,**

**child: VideoPlayer(\_controller),**

**)**

**: Container(),**

**),**

**floatingActionButton: FloatingActionButton(**

**onPressed: () {**

**setState(() {**

**\_controller.value.isPlaying**

**? \_controller.pause()**

**: \_controller.play();**

**});**

**},**

**child: Icon(**

**\_controller.value.isPlaying ? Icons.pause : Icons.play\_arrow,**

**),**

**),**

**);**

**}**

**@override**

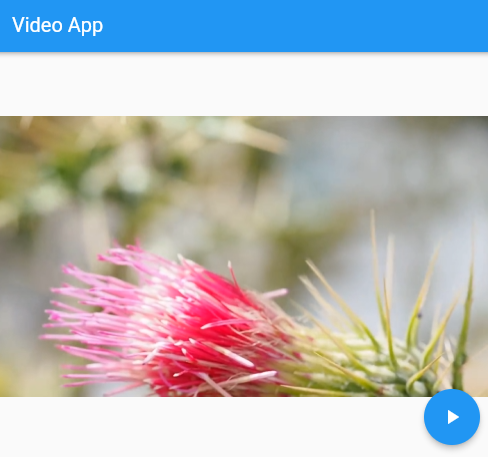
**void dispose() {**

**super.dispose();**

**\_controller.dispose();**

**}**

**}**

****

**Q13. Create the list of students and display it using the list view**

**Ans.**

**import 'package:flutter/material.dart';**

**void main() {**

**runApp(const MyApp());**

**}**

**class MyApp extends StatelessWidget {**

**const MyApp({super.key});**

**@override**

**Widget build(BuildContext context) {**

**return MaterialApp(**

**title: 'Flutter Demo',**

**debugShowCheckedModeBanner: false,**

**theme: ThemeData(**

**primarySwatch: Colors.blue,**

**),**

**home: const MyHomePage(),**

**);**

**}**

**}**

**class MyHomePage extends StatefulWidget {**

**const MyHomePage({super.key});**

**@override**

**State<MyHomePage> createState() => \_MyHomePageState();**

**}**

**class \_MyHomePageState extends State<MyHomePage> {**

**List<String> studentData = [**

**"Superman",**

**"Spider Man",**

**"Batman",**

**"Iron Man",**

**"Wonder Woman",**

**"Hulk",**

**];**

**@override**

**Widget build(BuildContext context) {**

**return Scaffold(**

**appBar: AppBar(**

**title: Text("App"),**

**),**

**body: SizedBox(**

**child: ListView.builder(**

**itemCount: studentData.length,**

**itemBuilder: (context, index) {**

**return Padding(**

**padding: const EdgeInsets.all(4.0),**

**child: SizedBox(**

**width: double.infinity,**

**height: 80,**

**child: Card(**

**elevation: 10,**

**child: Center(**

**child: Text(**

**studentData[index],**

**style: TextStyle(**

**fontSize: 21,**

**fontWeight: FontWeight.bold,**

**color: Colors.blue,**

**),**

**),**

**),**

**),**

**),**

**);**

**},**

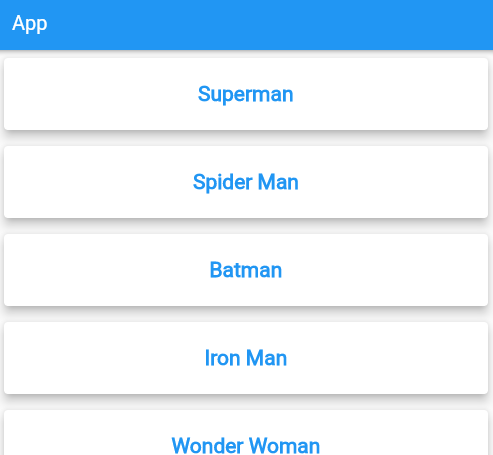
**),**

**),**

**);**

**}**

**}**

****

**Q14. Create an application to create the JSON File for storing user credentials and reading the stored data.**

**Ans.**

* Add the package to pubspec.yaml

path\_provider: ^2.0.11

**import 'dart:convert';**

**import 'dart:developer';**

**import 'dart:io';**

**import 'package:flutter/material.dart';**

**import 'package:path\_provider/path\_provider.dart';**

**void main() {**

**runApp(const MyApp());**

**}**

**class MyApp extends StatelessWidget {**

**const MyApp({super.key});**

**@override**

**Widget build(BuildContext context) {**

**return MaterialApp(**

**title: 'Flutter Demo',**

**debugShowCheckedModeBanner: false,**

**theme: ThemeData(**

**primarySwatch: Colors.blue,**

**),**

**home: const MyHomePage(),**

**);**

**}**

**}**

**class MyHomePage extends StatefulWidget {**

**const MyHomePage({super.key});**

**@override**

**State<MyHomePage> createState() => \_MyHomePageState();**

**}**

**class \_MyHomePageState extends State<MyHomePage> {**

**List<User?>? allUserData = [];**

**var ct1 = TextEditingController();**

**var ct2 = TextEditingController();**

**\_write() async {**

**String text = "";**

**try {**

**final Directory directory = await getApplicationDocumentsDirectory();**

**final File file = File('${directory.path}/user.json');**

**text = await file.readAsString();**

**var userdata = userFromJson(text);**

**userdata!.add(**

**User(username: ct1.text.trim(), password: ct2.text.trim()),**

**);**

**log("USER DATA $userdata");**

**file.writeAsString(userToJson(userdata));**

**getData();**

**} catch (e) {**

**print("Couldn't read file $e");**

**}**

**}**

**void getData() async {**

**final Directory directory = await getApplicationDocumentsDirectory();**

**final File file = File('${directory.path}/user.json');**

**String text = await file.readAsString();**

**setState(() {**

**allUserData = userFromJson(text);**

**});**

**}**

**@override**

**void initState() {**

**super.initState();**

**getData();**

**}**

**@override**

**Widget build(BuildContext context) {**

**log("INIT value $allUserData");**

**return Scaffold(**

**appBar: AppBar(**

**title: Text("App"),**

**),**

**body: SizedBox(**

**child: Column(**

**children: [**

**Padding(**

**padding: const EdgeInsets.all(8.0),**

**child: TextField(**

**controller: ct1,**

**decoration: InputDecoration(**

**labelText: "Username",**

**border: OutlineInputBorder(),**

**),**

**),**

**),**

**Padding(**

**padding: const EdgeInsets.all(8.0),**

**child: TextField(**

**controller: ct2,**

**decoration: InputDecoration(**

**labelText: "Password",**

**border: OutlineInputBorder(),**

**),**

**),**

**),**

**ElevatedButton(**

**onPressed: \_write,**

**child: Text(**

**"Write ",**

**),**

**),**

**Expanded(**

**child: ListView.builder(**

**itemCount: allUserData!.length,**

**itemBuilder: (context, index) {**

**return Card(**

**elevation: 8,**

**child: ListTile(**

**title: Text("Username: ${allUserData![index]!.username}"),**

**subtitle:**

**Text("Password: ${allUserData![index]!.password}"),**

**),**

**);**

**},**

**),**

**),**

**],**

**),**

**),**

**);**

**}**

**}**

**List<User?>? userFromJson(String str) => json.decode(str) == null**

**? []**

**: List<User?>.from(json.decode(str)!.map((x) => User.fromJson(x)));**

**String userToJson(List<User?>? data) => json.encode(**

**data == null ? [] : List<dynamic>.from(data.map((x) => x!.toJson())));**

**class User {**

**User({**

**required this.username,**

**required this.password,**

**});**

**String? username;**

**String? password;**

**User copyWith({**

**String? username,**

**String? password,**

**}) =>**

**User(**

**username: username ?? this.username,**

**password: password ?? this.password,**

**);**

**factory User.fromJson(Map<String, dynamic> json) => User(**

**username: json["username"],**

**password: json["password"],**

**);**

**Map<String, dynamic> toJson() => {**

**"username": username,**

**"password": password,**

**};**

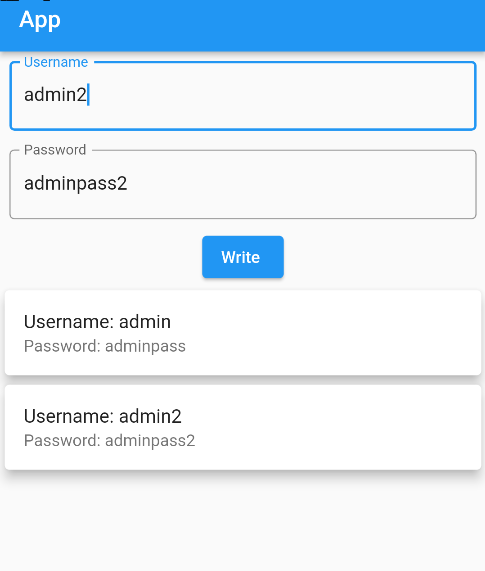
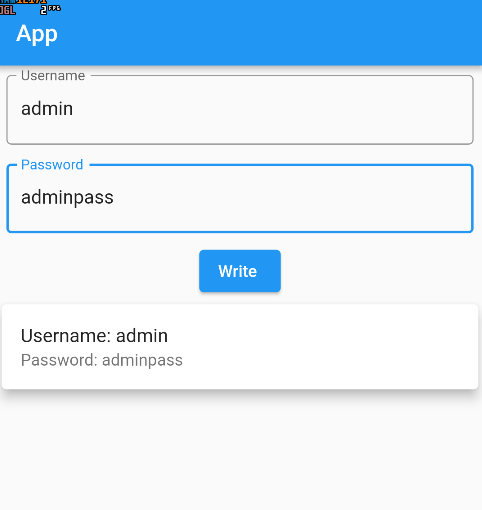
**@override**

**String toString() {**

**return "{ username : $username, password : $password }";**

**}**

**}**

****

**Q15. Create an application to store the data using SQLite database/ firebase database.**

**Ans.**

* **SQLite Database**
* Add the package to pubspec.yaml

**sqlite3: ^1.9.1**

**sqlite3\_flutter\_libs: ^0.5.12**

**import 'dart:convert';**

**import 'dart:developer';**

**import 'package:flutter/material.dart';**

**import 'package:sqlite3/sqlite3.dart';**

**void main() {**

**runApp(const MyApp());**

**}**

**class MyApp extends StatelessWidget {**

**const MyApp({super.key});**

**@override**

**Widget build(BuildContext context) {**

**return MaterialApp(**

**title: 'Flutter Demo',**

**debugShowCheckedModeBanner: false,**

**theme: ThemeData(**

**primarySwatch: Colors.blue,**

**),**

**home: const MyHomePage(),**

**);**

**}**

**}**

**class MyHomePage extends StatefulWidget {**

**const MyHomePage({super.key});**

**@override**

**State<MyHomePage> createState() => \_MyHomePageState();**

**}**

**class \_MyHomePageState extends State<MyHomePage> {**

**List<User?>? allUserData = [];**

**var ct1 = TextEditingController();**

**var ct2 = TextEditingController();**

**late Database db;**

**late ResultSet resultSet;**

**\_write() async {**

**final stmt =**

**db.prepare('INSERT INTO users (username,password) VALUES (?,?)');**

**stmt.execute([ct1.text.trim(), ct2.text.trim()]);**

**getData();**

**stmt.dispose();**

**try {} catch (e) {**

**print("Couldn't write in SQL $e");**

**}**

**}**

**void getData() async {**

**ResultSet resultSet = db.select('SELECT \* FROM users');**

**allUserData = [];**

**for (final row in resultSet) {**

**allUserData!**

**.add(User(username: row['username'], password: row['password']));**

**print(**

**'User[id: ${row['id']}, username: ${row['username']}, password: ${row['password']}]');**

**}**

**setState(() {});**

**}**

**@override**

**void initState() {**

**super.initState();**

**db = sqlite3.openInMemory();**

**db.execute('''**

**CREATE TABLE users (**

**id INTEGER NOT NULL PRIMARY KEY,**

**username TEXT NOT NULL,**

**password TEXT NOT NULL**

**);**

**''');**

**getData();**

**}**

**@override**

**Widget build(BuildContext context) {**

**log("INIT value $allUserData");**

**return Scaffold(**

**appBar: AppBar(**

**title: Text("App"),**

**),**

**body: SizedBox(**

**child: Column(**

**children: [**

**Padding(**

**padding: const EdgeInsets.all(8.0),**

**child: TextField(**

**controller: ct1,**

**decoration: InputDecoration(**

**labelText: "Username",**

**border: OutlineInputBorder(),**

**),**

**),**

**),**

**Padding(**

**padding: const EdgeInsets.all(8.0),**

**child: TextField(**

**controller: ct2,**

**decoration: InputDecoration(**

**labelText: "Password",**

**border: OutlineInputBorder(),**

**),**

**),**

**),**

**ElevatedButton(**

**onPressed: \_write,**

**child: Text(**

**"Write ",**

**),**

**),**

**Expanded(**

**child: ListView.builder(**

**itemCount: allUserData!.length,**

**itemBuilder: (context, index) {**

**return Card(**

**elevation: 8,**

**child: ListTile(**

**title: Text("Username: ${allUserData![index]!.username}"),**

**subtitle:**

**Text("Password: ${allUserData![index]!.password}"),**

**),**

**);**

**},**

**),**

**),**

**],**

**),**

**),**

**);**

**}**

**}**

**List<User?>? userFromJson(String str) => json.decode(str) == null**

**? []**

**: List<User?>.from(json.decode(str)!.map((x) => User.fromJson(x)));**

**String userToJson(List<User?>? data) => json.encode(**

**data == null ? [] : List<dynamic>.from(data.map((x) => x!.toJson())));**

**class User {**

**User({**

**required this.username,**

**required this.password,**

**});**

**String? username;**

**String? password;**

**User copyWith({**

**String? username,**

**String? password,**

**}) =>**

**User(**

**username: username ?? this.username,**

**password: password ?? this.password,**

**);**

**factory User.fromJson(Map<String, dynamic> json) => User(**

**username: json["username"],**

**password: json["password"],**

**);**

**Map<String, dynamic> toJson() => {**

**"username": username,**

**"password": password,**

**};**

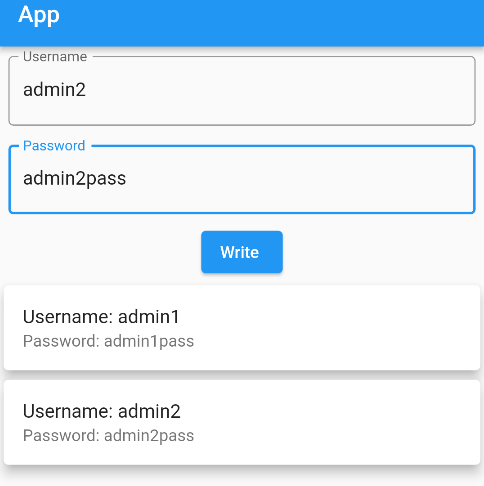
**@override**

**String toString() {**

**return "{ username : $username, password : $password }";**

**}**

**}**

****