Name - jatin talreja Div - D15A ,Roll No - 61 ,Batch - c

Experiment No -04 Exploring Form Widgets

Aim - To create interactive forms using form widgets.

Theory -

Obscure Text -the obscureText property is often used in text input fields like TextFormField to obscure the entered text, typically for password fields or any other sensitive information.

TextField: The TextField widget is used to collect user input via text entry. It supports various input types, such as text, numbers, passwords, and more, and provides options for customization and validation.

HintStyle - The style of the hint text displayed in form fields using the hintStyle property of the InputDecoration class. This property allows you to define the text style for the hint text, including its color, font size, font weight, and more.

TextInputType - This property is used to indicate to the system what type of input is expected from the user, and it adjusts the keyboard layout accordingly.

1.TextInputType.text: The default keyboard type for general text input.

2.TextInputType.number: A keyboard optimized for numeric input. It typically includes digits, a decimal point, and symbols for arithmetic operations.

3.**TextInputType.phone**: A keyboard optimized for entering phone numbers. It may include digits, plus sign, and symbols commonly used in phone numbers.

4.**TextInputType.emailAddress**: A keyboard optimized for entering email addresses. It includes characters commonly used in email addresses, such as "@" and ".".

Code -

```
import 'dart:convert';

import 'package:flutter/material.dart';
import 'package:intl/intl.dart';
import 'package:http/http.dart' as http;

import '../Home/home.dart';


final formatter = DateFormat.yMd();
 var enteredname='';
 var entereddate;

class SignUp extends StatefulWidget {
  const SignUp({super.key});
  @override
  State<StatefulWidget> createState() {
    return _SignUp();
  }
}
class DT{
DT({required this.date});
final DateTime date;
String get formattefDate {
    return formatter.format(date);
  }

}



class _SignUp extends State<SignUp> {
  DateTime? _selectedDate=DateTime.now();
  void _presentDatePicker() async {
```

```
    final now = DateTime.now();
    final firstDate = DateTime(now.year - 1, now.month, now.day);
    // final lastDate = DateTime(now.year + 3, now.month, now.day);
    final pickedDate = await showDatePicker(
        context: context,
        initialDate: now,
        firstDate: firstDate,
        lastDate: now);
     if(_selectedDate!=null){
    setState(() {
      _selectedDate = pickedDate??_selectedDate;
    });}
  }
   String get formattedDate {
    return formatter.format(DateTime.now());
  }

  void savedata()async{

  enteredname=nameController.text;
  final
url=Uri.https('flutter-prep-5b74d-default-rtdb.firebaseio.com','user-data.json'
);

    final response= await http.post(url,headers: {
        'Content-type':'application/json'
        },
      body: json.encode({
        'name': enteredname,
        'dateofbirth': formatter.format(_selectedDate!)
    })
      );

}
```

```
final _dateController = TextEditingController();
final nameController=TextEditingController();

@override
Widget build(BuildContext context) {
  DateTime? selectedDate=DateTime.now();
  // var selectedDate = _selectedDate;

  return Scaffold(
    backgroundColor: Theme.of(context).primaryColorLight,
    body:  Center(
      child: Padding(
        padding: const EdgeInsets.all(12.0),
        child: Column(
          mainAxisSize: MainAxisSize.min,
          mainAxisAlignment: MainAxisAlignment.center,
          crossAxisAlignment: CrossAxisAlignment.center,
          children: [
            TextField(
              controller: nameController,
              keyboardType: TextInputType.name,
              decoration: InputDecoration(
                border: OutlineInputBorder(
                    borderRadius: BorderRadius.all(Radius.circular(8))),
                iconColor: Colors.black,
                label: Text(
                  "Add Your Name",
                  style: TextStyle(
                    color: Colors.black,
                    fontStyle: FontStyle.italic,
                  ),
                ),
              ),
              expands: false,
            ),
```

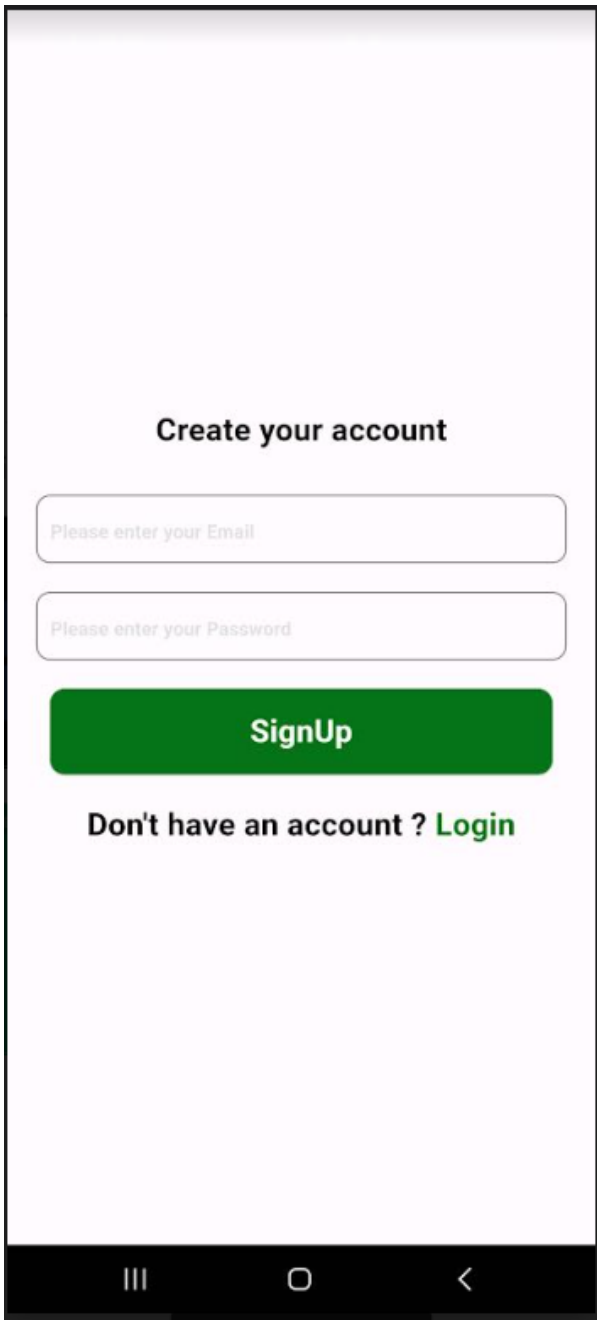
```
const SizedBox(height: 10,),
const SizedBox(height: 10),
Row(
  children: [
     Expanded(
       child: TextField(
         controller: _dateController,
         decoration: InputDecoration(
           label: Text('Enter your D.O.B',style: TextStyle(
         color: Colors.black,
         fontStyle: FontStyle.italic,
       ),
       )
         ),
       ),
     ),
       Expanded(
     child: Row(
       mainAxisAlignment: MainAxisAlignment.end,
       crossAxisAlignment: CrossAxisAlignment.center,
       children: [
         Text(

        selectedDate == null
              ? 'No Date Selected'
              : formatter.format(_selectedDate!),
              style: const TextStyle(color: Colors.black)
         ),
         const SizedBox(width: 10),
         IconButton(
            onPressed: () {
              _presentDatePicker();
            },
            icon: const Icon(
              Icons.calendar_month,
```

```
                    color: Colors.black,
                  )),
              // const SizedBox(width: 16),
              // Text(formattedDate),
            ],
          ),
        ),

      ],
    ),
    const SizedBox(height: 10,),
    ElevatedButton(onPressed: (){
      savedata();
      Navigator.of(context).push(MaterialPageRoute(builder: (ctx)=>
          MyHomePage())
      );
    }, child: const Text("Next")),
  ],
 ),
),
),
);
}
}
```

Output -



Conclusion - in this experiment we learnt about forms creation