Flutter Assignment

Q1. Create a small To-Do List screen in Flutter where users can add

- **tasks and mark them as completed.**
- **AppBar title: My Todos**
- **Add Task - TextField to enter task and Add button next to it**
- **When the user types text and presses Add:**
- **If the text is not empty, add it to the todo list and Clear the TextField after adding**
- **If the text is empty, do nothing**

```
import 'package:flutter/material.dart';
import 'package:flutter_application_1/AppScaffold.dart';
import 'package:flutter_application_1/Todos.dart';
class Question1 extends StatefulWidget {
  const Question1({super.key});
  @override
  State<Question1> createState() => _Question1State();
}
class _Question1State extends State<Question1> {
  final TextEditingController _task = TextEditingController();
  List<Todos> todoList = [];
  @override
  Widget build(BuildContext context) {
    return AppScaffold(
      title: "My Todo App",
      child: Padding(
        padding: const EdgeInsets.all(20),
        child: Column(
          children: [
```

```
const Text(
  "Todo App",
  style: TextStyle(fontSize: 24, fontWeight: FontWeight.bold),
),
const SizedBox(height: 20),
Row(
  children: [
    Expanded(
      child: TextField(
        controller: _task,
        decoration: InputDecoration(
          border: OutlineInputBorder(),
          labelText: 'Enter your todo',
          hintText: 'e.g. Buy groceries',
        ),
      ),
    ),
    const SizedBox(width: 10),
    ElevatedButton(
      onPressed: () {
        if (_task.text.isEmpty) {
          ScaffoldMessenger.of(context).showSnackBar(
            const SnackBar(content: Text('Please enter a task')),
          );
          return;
        }
        setState(() {
          todoList.add(Todos(task: _task.text));
```

```
          _task.clear();
        });
      },
      child: const Text('Add'),
    ),
  ],
),
const SizedBox(height: 20),
Expanded(
  child: Stack(
    children: [
      ListView.builder(
        itemCount: todoList.length,
        itemBuilder: (context, index) {
          return ListTile(
            title: Text(
              todoList[index].task,
              style: TextStyle(
                decoration: todoList[index].isComplete
                    ? TextDecoration.lineThrough
                    : TextDecoration.none,
                color: todoList[index].isComplete
                    ? Colors.grey
                    : Colors.black,
                fontSize: 20,
              ),
            ),
```

```
                    leading: Checkbox(
                      value: todoList[index].isComplete,
                      onChanged: (value) {
                        setState(() {
                          todoList[index].isComplete =
                              !todoList[index].isComplete;
                        });
                      },
                    ),
                  );
                },
              ),
            ],
          ),
        ),
      ],
    ),
  ),
 );
 }
}
class Todos {
 String task;
 bool isComplete;
 Todos({required this.task, this.isComplete = false});
}
```

Q2. Create a Flutter app with two screens: Login and Welcome.

- **Screen 1: Login**
  - **Use a TextFormField for Email and Password**
  - **Add a Checkbox labeled "Remember me"**
  - **Add a large indigo backroundcolor "Login" button**
  - **If both email and password are not empty → navigate to Welcome screen**
  - **If any field is empty → show SnackBar "Fill both fields"**
- **Screen 2: Welcome**
  - **AppBar title: "Welcome"**
  - **Display big text: "Hello, User!"**

```
import 'package:flutter/material.dart';
import 'package:flutter_application_1/AppScaffold.dart';
import 'package:flutter_application_1/Login.dart';
class Question2 extends StatefulWidget {
  const Question2({super.key});
  @override
  State<Question2> createState() => _Question2State();
}
class _Question2State extends State<Question2> {
  @override
  Widget build(BuildContext context) {
    return AppScaffold(title: "Question 2", child: Padding(
      padding: const EdgeInsets.all(20),
      child: Container(
        alignment: Alignment.center,
        child: Login(),
      ),
    ));
  }
}
```

```
import 'package:flutter/material.dart';
import 'package:flutter_application_1/Dashboard.dart';
class Login extends StatefulWidget {
  const Login({super.key});
  @override
  State<Login> createState() => _LoginState();
}
class _LoginState extends State<Login> {
  final GlobalKey<FormState> _formKey = GlobalKey<FormState>();
  TextEditingController email = TextEditingController();
  TextEditingController password = TextEditingController();
  bool obscureText = true;
  @override
  Widget build(BuildContext context) {
    return Container(
      alignment: Alignment.center,
      child: Column(
        children: [
          Form(
            key: _formKey,
            child: Column(
              children: [
                Text(
                  "Login",
                  style: TextStyle(fontSize: 24, fontWeight: FontWeight.bold),
                ),

                SizedBox(height: 20),
```

```
TextFormField(
  controller: email,
  decoration: InputDecoration(
    labelText: "Email",
    hintText: "Enter your email",
    border: OutlineInputBorder(),
  ),
  validator: (value) {
    String pattern =
        r'^[\w-]+(\.[\w-]+)*@([\w-]+\.)+[a-zA-Z]{2,7}$';
    r'^[\w-]+(\.[\w-]+)*@([\w-]+\.)+[a-zA-Z]{2,7}$';
    if (value == null || value.isEmpty) {
      return 'Please enter your email';
    } else if (!RegExp(pattern).hasMatch(value)) {
      return 'Please enter a valid email address';
    }
    return null;
  },
),
SizedBox(height: 20),
TextFormField(
  obscureText: obscureText,
  controller: password,
  decoration: InputDecoration(
    suffixIcon: IconButton(
      icon: Icon(
        obscureText ? Icons.visibility_off : Icons.visibility,
```

```
      ),
      onPressed: () {
        setState(() {
          obscureText = !obscureText;
        });
      },
    ),
    border: OutlineInputBorder(),
    labelText: "Password",
    hintText: "Enter your password",
  ),
  validator: (value) {
    if (value == null || value.isEmpty) {
      return 'Please enter your password';
    } else if (value.length < 8 || value.length > 20) {
      return "Password min 8 char and max 20 char";
    }
    return null;
  },
),
SizedBox(height: 20),
SizedBox(
  width: double.infinity,
  height: 45,
  child: ElevatedButton(
    onPressed: () {
      if (_formKey.currentState!.validate()) {
        Navigator.pushReplacement(
```

```
                context,
                MaterialPageRoute(
                  builder: (context) => Dashboard(email: email.text),
                ),
              );
            }
          },
          style: ElevatedButton.styleFrom(
            foregroundColor: Colors.white,
            backgroundColor: Colors.indigo,
            shape: RoundedRectangleBorder(
              borderRadius: BorderRadius.circular(10),
            ),
          ),
          child: Text("Login"),
        ),
      ),
    ],
  ),
 ),
 ],
 ),
 ),
 ),
 );
 }
}
```

Q3. Develop a Flutter application screen that displays a list of items using ListView. The screen must include a Floating Action Button which, when pressed, dynamically adds a new item to the list and updates the UI immediately.

1. **Display a list of items on the screen.**
2. **Use ListView or ListView.builder to show the list.**
3. **Add a FloatingActionButton at the bottom-right corner.**
4. **On clicking the FAB: A new item should be added to the list.**

```
import 'package:flutter/material.dart';
class Question3 extends StatefulWidget {
  const Question3({super.key});
  @override
  State<Question3> createState() => _Question3State();
}
class _Question3State extends State<Question3> {
  List<String> items = List.generate(10, (index) => "Item ${index + 1}");
  final ScrollController _scrollController = ScrollController();
  @override
  void dispose() {
    _scrollController.dispose();
    super.dispose();
  }
  @override
  Widget build(BuildContext context) {
    return Scaffold(
      floatingActionButton: FloatingActionButton(
        onPressed: () {
          setState(() {
```

```
      items.add("Item ${items.length + 1}");
    });
    WidgetsBinding.instance.addPostFrameCallback((_) {
      if (!_scrollController.hasClients) {
        return;
      }
      _scrollController.jumpTo(
        _scrollController.position.maxScrollExtent,
      );
    });
  },
  child: Icon(Icons.add),
),
appBar: AppBar(
  title: Text("Question 3"),
  backgroundColor: Colors.blue,
  foregroundColor: Colors.white,
),
body: Stack(
  children: [
    ListView.builder(
      controller: _scrollController,
      itemCount: items.length,
      itemBuilder: (context, index) {
        return ListTile(title: Text(items[index]));
      },
    ),
  ],
```

```
      ),
    );
  }
}
```

Q4. Design a Flutter Login Screen with proper validation and background design.

- **Email input using TextFormField - Field is required and mail must be**
- **in valid format**
- **Password input using TextFormField - Minimum 8 characters**
- **Add show/hide password option**
- **Submit button**

```
import 'package:flutter/material.dart';
import 'package:flutter_application_1/Dashboard.dart';
class Login extends StatefulWidget {
  const Login({super.key});
  @override
  State<Login> createState() => _LoginState();
}
class _LoginState extends State<Login> {
  final GlobalKey<FormState> _formKey = GlobalKey<FormState>();
  TextEditingController email = TextEditingController();
  TextEditingController password = TextEditingController();
  bool obscureText = true;
  @override
  Widget build(BuildContext context) {
    return Container(
      alignment: Alignment.center,
      child: Column(
        children: [
          Form(
            key: _formKey,
            child: Column(
```

```
children: [
 Text(
  "Login",
  style: TextStyle(fontSize: 24, fontWeight: FontWeight.bold),
 ),
 SizedBox(height: 20),
 TextFormField(
  controller: email,
  decoration: InputDecoration(
   labelText: "Email",
   hintText: "Enter your email",
   border: OutlineInputBorder(),
  ),
  validator: (value) {
   String pattern =
       r'^[\w-]+(\.[\w-]+)*@([\w-]+\.)+[a-zA-Z]{2,7}$';
   r'^[\w-]+(\.[\w-]+)*@([\w-]+\.)+[a-zA-Z]{2,7}$';
   if (value == null || value.isEmpty) {
    return 'Please enter your email';
   } else if (!RegExp(pattern).hasMatch(value)) {
    return 'Please enter a valid email address';
   }
   return null;
  },
 ),

 SizedBox(height: 20),
```

```
TextFormField(
  obscureText: obscureText,
  controller: password,
  decoration: InputDecoration(
    suffixIcon: IconButton(
      icon: Icon(
        obscureText ? Icons.visibility_off : Icons.visibility,
      ),
      onPressed: () {
        setState(() {
          obscureText = !obscureText;
        });
      },
    ),
    border: OutlineInputBorder(),
    labelText: "Password",
    hintText: "Enter your password",
  ),
  validator: (value) {
    if (value == null || value.isEmpty) {
      return 'Please enter your password';
    } else if (value.length < 8 || value.length > 20) {
      return "Password min 8 char and max 20 char";
    }
    return null;
  },
),
```

```
      SizedBox(height: 20),
      SizedBox(
        width: double.infinity,
        height: 45,
        child: ElevatedButton(
          onPressed: () {
            if (_formKey.currentState!.validate()) {
              Navigator.pushReplacement(
                context,
                MaterialPageRoute(
                  builder: (context) => Dashboard(email: email.text),
                ),
              );
            }
          },
          style: ElevatedButton.styleFrom(
            foregroundColor: Colors.white,
            backgroundColor: Colors.indigo,
            shape: RoundedRectangleBorder(
              borderRadius: BorderRadius.circular(10),
            ),
          ),
          child: Text("Login"),
        ),
      ),
    ],
  ),
),
```

```
        ],
      ),
    );
  }
}
```

Q5. Create a Flutter app with two screens to add books and view the book list.

- **Screen 1: Add Book**
  - **TextField for Book Title and Author Name**
  - **Add Book button - On button presses -> If both fields are not empty → add book to list and go to View screen**
- **Screen 2: View Books**
  - **AppBar title: "My Books"**
  - **Display books using ListView**
  - **Each row shows: Book Title, Author Name**

```
import 'package:flutter/material.dart';
import 'package:flutter_application_1/AddBook.dart';
import 'package:flutter_application_1/AppScaffold.dart';
import 'package:flutter_application_1/Books.dart';
import 'package:flutter_application_1/ListBooks.dart';
class Question5 extends StatefulWidget {
  const Question5({super.key});
  @override
  State<Question5> createState() => _Question5State();
}
class _Question5State extends State<Question5> {
  List<Map<String, String>> book = Books().displayBooks();
  Books books = Books();
  @override
  Widget build(BuildContext context) {
    return AppScaffold(
      title: "Question 5",
      child: Padding(
        padding: const EdgeInsets.all(20),
```

```
child: Column(
  children: [
    Row(
      mainAxisAlignment: MainAxisAlignment.spaceBetween,
      children: [
        ElevatedButton(
          onPressed: () {
            Navigator.push(
              context,
              MaterialPageRoute(builder: (context) => AddBook(books: books,)),
            );
          },
          child: Text("Add Book"),
        ),
        ElevatedButton(
          onPressed: () {
            Navigator.push(
              context,
              MaterialPageRoute(builder: (context) => Listbooks(books: books,)),
            );
          },
          child: Text("View Books"),
        ),
      ],
    ),
    SizedBox(height: 20),
  ],
),
```

```
      ),
    );
  }
}
import 'package:flutter/material.dart';
import 'package:flutter_application_1/AppScaffold.dart';
import 'package:flutter_application_1/Books.dart';
import 'package:flutter_application_1/ListBooks.dart';
class AddBook extends StatefulWidget {
  final Books books;
  const AddBook({super.key, required this.books});
  @override
  State<AddBook> createState() => _AddBookState();
}
class _AddBookState extends State<AddBook> {
  TextEditingController titleController = TextEditingController();
  TextEditingController authorController = TextEditingController();
  @override
  Widget build(BuildContext context) {
    return AppScaffold(
      title: "Add Book",
      child: Padding(
        padding: const EdgeInsets.all(20),
        child: Column(
          crossAxisAlignment: CrossAxisAlignment.start,
          mainAxisAlignment: MainAxisAlignment.start,
          children: [
            Text("Title", style: TextStyle(fontWeight: FontWeight.bold)),
```

```
TextField(
  controller: titleController,
  decoration: InputDecoration(
    hintText: "Enter book title",
    border: OutlineInputBorder(),
  ),
),
SizedBox(height: 20),
Text("Author", style: TextStyle(fontWeight: FontWeight.bold)),
TextField(
  controller: authorController,
  decoration: InputDecoration(
    hintText: "Enter author name",
    border: OutlineInputBorder(),
  ),
),
SizedBox(height: 20),
ElevatedButton(
  onPressed: () {
    if (titleController.text.isEmpty ||
        authorController.text.isEmpty) {
      ScaffoldMessenger.of(context).showSnackBar(
        SnackBar(content: Text("Please fill in all fields")),
      );
      return;
    }
    widget.books.addBook(
      titleController.text,
```

```
              authorController.text,
            );
            Navigator.pushReplacement(
              context,
              MaterialPageRoute(
                builder: (context) => Listbooks(books: widget.books),
              ),
            );
          },
          child: Text("Add Book"),
        ),
      ],
    ),
  ),
 );
 }
}
import 'package:flutter/material.dart';
import 'package:flutter_application_1/AppScaffold.dart';
import 'package:flutter_application_1/Books.dart';
class Listbooks extends StatefulWidget {
 final Books books;
 const Listbooks({super.key, required this.books});
 @override
 State<Listbooks> createState() => _ListbooksState();
}
class _ListbooksState extends State<Listbooks> {
 List<Map<String, String>> book = Books().displayBooks();
```

```
  @override
  Widget build(BuildContext context) {
    return AppScaffold(
      title: "View Books",
      child: Stack(
        children: [
          ListView.builder(
            itemCount: widget.books.displayBooks().length,
            itemBuilder: (context, index) {
              return ListTile(
                title: Text(widget.books.displayBooks()[index]["title"] ?? ""),
                subtitle: Text(widget.books.displayBooks()[index]["author"] ?? ""),
              );
            },
          ),
        ],
      ),
    );
  }
}
```

```
class Books {
  String? title;
  String? author;
  static List<Map<String, String>> bookList = [{
    "title": "Flutter for Beginners",
    "author": "John Doe"
  }, {
    "title": "Dart in Action",
    "author": "Jane Smith"
  }];
  // Books({required this.title, required this.author});
  void addBook(String title, String author) {
    bookList.add({"title": title, "author": author});
  }
  List<Map<String, String>> displayBooks() {
    return bookList;
  }
}
```

Q6. Design a Product Catalog screen using GridView.

- **Display products in a 2-column GridView**
- **Each product card must show: Product image, Product name and Price**
- **Store product data using a List<Map>**

```
import 'package:flutter/material.dart';
import 'package:flutter_application_1/AppScaffold.dart';
class Product extends StatefulWidget {
  const Product({super.key});
  @override
  State<Product> createState() => _ProductState();
}
class _ProductState extends State<Product> {
  List<Map<String, dynamic>> products = [
    {
      "product_image": "https://placehold.co/400/png",
      "product_name": "Example Product 1",
      "price": 100,
    },
    {
      "product_image": "https://placehold.co/400/png",
      "product_name": "Example Product 2",
      "price": 78,
    },
    {
      "product_image": "https://placehold.co/400/png",
      "product_name": "Example Product 3",
      "price": 654,
```

```
  },
  {
    "product_image": "https://placehold.co/400/png",
    "product_name": "Example Product 4",
    "price": 12,
  },
  {
    "product_image": "https://placehold.co/400/png",
    "product_name": "Example Product 5",
    "price": 89,
  },
  {
    "product_image": "https://placehold.co/400/png",
    "product_name": "Example Product 6",
    "price": 2000,
  },
  {
    "product_image": "https://placehold.co/400/png",
    "product_name": "Example Product 7",
    "price": 10000,
  },
];
@override
Widget build(BuildContext context) {
  return AppScaffold(
    title: "Product Catalog",
    child: Padding(
      padding: EdgeInsets.all(20),
```

```
child: Column(
  mainAxisAlignment: MainAxisAlignment.start,
  crossAxisAlignment: CrossAxisAlignment.start,
  children: [
    Text(
      "Product Catalog",
      style: TextStyle(fontSize: 24, fontWeight: FontWeight.bold),
    ),
    Expanded(
      child: GridView.builder(
        gridDelegate: SliverGridDelegateWithFixedCrossAxisCount(
          crossAxisCount: 2,
          crossAxisSpacing: 2,
          mainAxisSpacing: 2,
          childAspectRatio: 0.75,
        ),
        itemCount: products.length,
        itemBuilder: (context, index) {
          return Card(
            child: Column(
              children: [
                Expanded(
                  child: Image.network(
                    products[index]["product_image"],
                    fit: BoxFit.cover,
                  ),
                ),
                SizedBox(height: 10),
```

```
                    Text(
                      products[index]["product_name"],
                      style: TextStyle(fontWeight: FontWeight.bold),
                    ),
                    SizedBox(height: 5),
                    Text("\$${products[index]["price"]}"),
                    SizedBox(height: 10),
                  ],
                ),
              );
            },
          ),
        ),
      ],
    ),
  ),
 );
 }
}
```

Q7. Display employees in a grid layout.

Each grid item shows:

- **Circular profile image**
- **Employee name**
- **Department**
- **On tap: Show employee Name in a SnackBar**

```
import 'package:flutter/material.dart';
import 'package:flutter_application_1/AppScaffold.dart';

class Employees extends StatefulWidget {
  const Employees({super.key});

  @override
  State<Employees> createState() => _EmployeesState();
}

class _EmployeesState extends State<Employees> {
  List<Map<String, String>> employees = [
    {"name": "Alice Johnson", "position": "Software Engineer"},
    {"name": "Bob Smith", "position": "Product Manager"},
    {"name": "Charlie Davis", "position": "UX Designer"},
  ];
  @override
  Widget build(BuildContext context) {
    return AppScaffold(
      title: "Employees",
      child: Padding(
        padding: EdgeInsets.all(20),
```

```
child: Column(
  children: [
    Text(
      "Employee List",
      style: TextStyle(fontSize: 24, fontWeight: FontWeight.bold),
    ),
    SizedBox(height: 20),
    Expanded(
      child: GridView.builder(
        gridDelegate: SliverGridDelegateWithMaxCrossAxisExtent(
          maxCrossAxisExtent: double.infinity,
          mainAxisExtent: 80,
        ),
        itemCount: employees.length,
        itemBuilder: (context, index) {
          final employee = employees[index];
          return ListTile(
            onTap: () => ScaffoldMessenger.of(context).showSnackBar(
              SnackBar(
                content: Text(
                  "${employee["name"]} - ${employee["position"]}",
                ),
              ),
            ),
            leading: CircleAvatar(child: Text(employee["name"]![0])),
            title: Text(employee["name"]!),
            subtitle: Text(employee["position"]!),
          );
```

```
            },
          ),
        ),
      ],
    ),
  ),
 );
 }
}
```