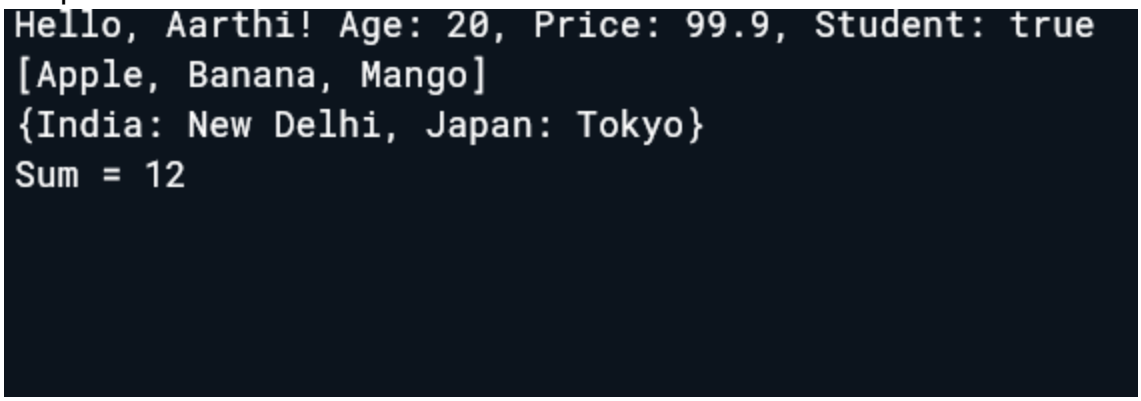
Flutter programs list for Mid-1

1.a.Write a simple Dart program to understand the language basics

Code:

```
void main() {
  int age = 20;
  String name = "Aarthi";
  double price = 99.9;
  bool isStudent = true;

  print("Hello, $name! Age: $age, Price: $price, Student: $isStudent");

  List<String> fruits = ["Apple", "Banana", "Mango"];
  print(fruits);

  Map<String, String> capitals = {"India": "New Delhi", "Japan": "Tokyo"};
  print(capitals);

  print("Sum = ${add(5, 7)}");
}

int add(int a, int b) {
  return a + b;
}
```

Output:



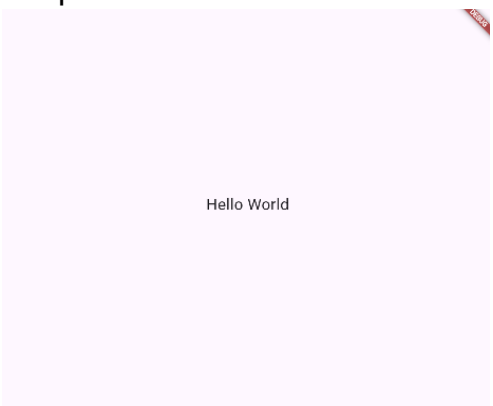
b) Write simple Flutter app to displays “Hello World” in the middle of the screen.

Code:

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

void main() {
  runApp(const MyApp());
}
```

```
class MyApp extends StatelessWidget {
  const MyApp({super.key});

  @override
  Widget build(BuildContext context) {
    return const MaterialApp(
      home: Scaffold(
        body: Center(
          child: Text(
            'Hello World',
            style: TextStyle(fontSize: 24),
          ),
        ),
      ),
    );
  }
}
```

Output:



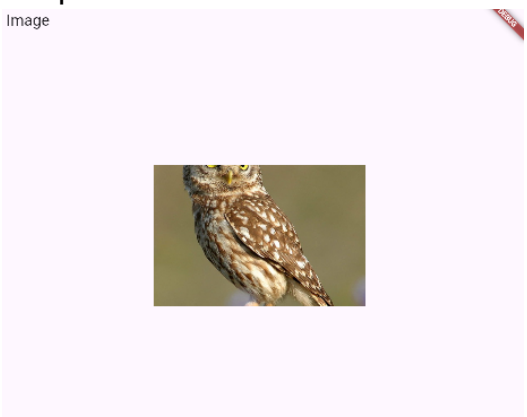
2.a) Explore various Flutter widgets (Text, Image, Container, etc.).

Code:

```
import 'package:flutter/material.dart';
void main() {
runApp(MyApp());
}
class MyApp extends StatefulWidget
const MyApp({super.key});
@override
State<MyApp> createState() => _MyAppState();
}
class _MyAppState extends State<MyApp> {
@override
```

```
Widget build(BuildContext context) {
return MaterialApp(
home: Scaffold(
appBar: AppBar(
title: Text('Image'),
),
body: Center(
child: Container(
height: 200,
width: 300,
child: Image.network(
'https://flutter.github.io/assets-for-api-docs/assets/widgets/owl.jpg',
//      valid image URL
fit: BoxFit.cover, // Optional: scales image to fit the container),
),

),

),

);

}

}
```

Output:



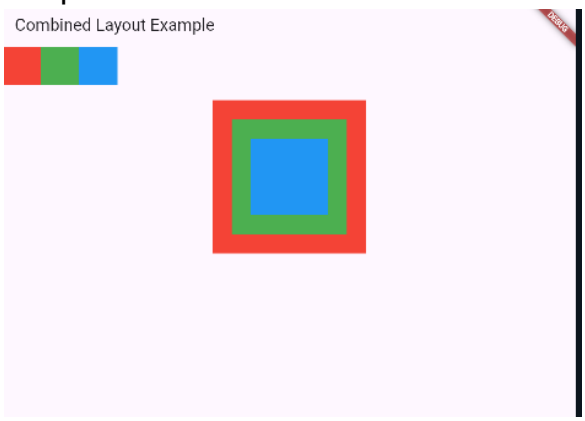
b) Implement different layout structures using Row, Column, and Stack widgets.

Code:

```
import 'package:flutter/material.dart';
void main() { runApp(MyApp()); }
class MyApp extends StatelessWidget {
```

```
@override
Widget build(BuildContext context) {
return MaterialApp( home: Scaffold(
appBar: AppBar(title: Text('Combined Layout Example')),
body: Center(child: Column(children: <Widget>

[Row(children: <Widget>[
Container(color: Colors.red, width: 50, height: 50),
Container(color: Colors.green, width: 50, height: 50),
Container(color: Colors.blue, width: 50, height: 50),], ),
SizedBox(height: 20),
Stack(alignment: Alignment.center,
children: <Widget>[Container(color: Colors.red, width: 200, height: 200),
Container(color: Colors.green, width: 150, height: 150),
Container(color: Colors.blue, width: 100, height: 100),], ),],
),
),
), ); } }
```

Output:



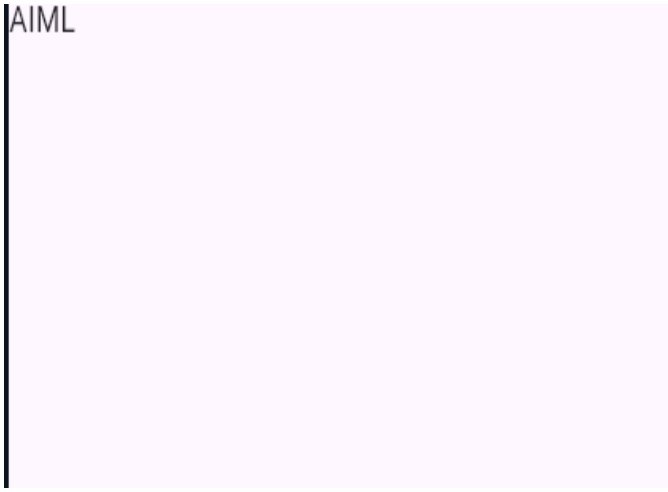
3.a) Design a responsive UI that adapts to different screen sizes.

Code:

```
import 'package:flutter/material.dart';
void main() { runApp(MyApp()); }
class MyApp extends StatelessWidget {
const MyApp({super.key});
@override
Widget build(BuildContext context) {
return MaterialApp(
home: SafeArea(
child:Scaffold(
body: Text('AIML'),
),
),);
```

}}

Output:



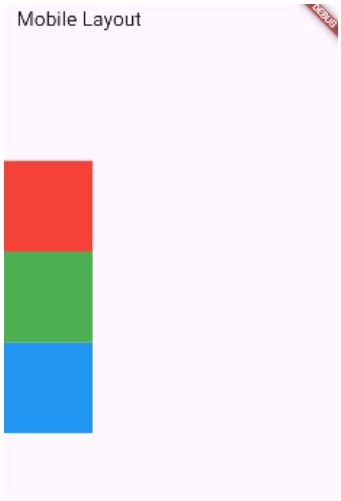
b) Implement media queries and breakpoints for responsiveness.

Code:

```
import 'package:flutter/material.dart';
void main() { runApp(MyApp()); }
class MyApp extends StatelessWidget {
Widget build(BuildContext context) {
return MaterialApp(home: ResponsiveLayout(), );
} }
class ResponsiveLayout extends StatelessWidget {
@override
Widget build(BuildContext context) {
var mediaQueryData = MediaQuery.of(context);
var screenWidth = mediaQueryData.size.width;
if (screenWidth < 600) {
return Scaffold(
appBar: AppBar(title: Text('Mobile Layout')),
body: _buildNarrowContainers(), );
} else if (screenWidth < 1200) {
return Scaffold(
appBar: AppBar(title: Text('Tablet Layout')),
body: _buildMediumContainers(),
);
} else {
return Scaffold(
appBar: AppBar(title: Text('Desktop Layout')),
body: _buildWideContainers(), );
} }
```

```
Widget _buildNarrowContainers() {
return Column(
mainAxisAlignment: MainAxisAlignment.center,
children: <Widget>[
Container(color: Colors.red, width: 100, height: 100),
Container(color: Colors.green, width: 100, height: 100),
Container(color: Colors.blue, width: 100, height: 100),], );
}
Widget _buildMediumContainers() {
return Row(
mainAxisAlignment: MainAxisAlignment.center,
children: <Widget>[
Container(color: Colors.red, width: 100, height: 100),
Container(color: Colors.green, width: 100, height: 100),
Container(color: Colors.blue, width: 100, height: 100), ], );
}
Widget _buildWideContainers() {
return GridView.count( crossAxisCount: 3,
mainAxisSpacing: 10,crossAxisSpacing: 10,
children: <Widget>[
Container(color: Colors.red, width: 100, height: 100),
Container(color: Colors.green, width: 100, height: 100),
Container(color: Colors.blue, width: 100, height: 100), ], );
} }
```

Output:



4.a) Implement Stateless widget

Code:

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

void main() {
  runApp(const MyApp());
}

// Custom Stateless Widget
class MyApp extends StatelessWidget {
  const MyApp({super.key});

  @override
  Widget build(BuildContext context) {
    return const MaterialApp(
      home: HomeScreen(),
    );
  }
}
```

```
class HomeScreen extends StatelessWidget {
  const HomeScreen({super.key});

  @override
  Widget build(BuildContext context) {
    return const Scaffold(
      body: Center(
        child: Text(
          "Hello from Stateless Widget!",
          style: TextStyle(fontSize: 24),
        ),
      ),
    );
  }
}
```

Output:

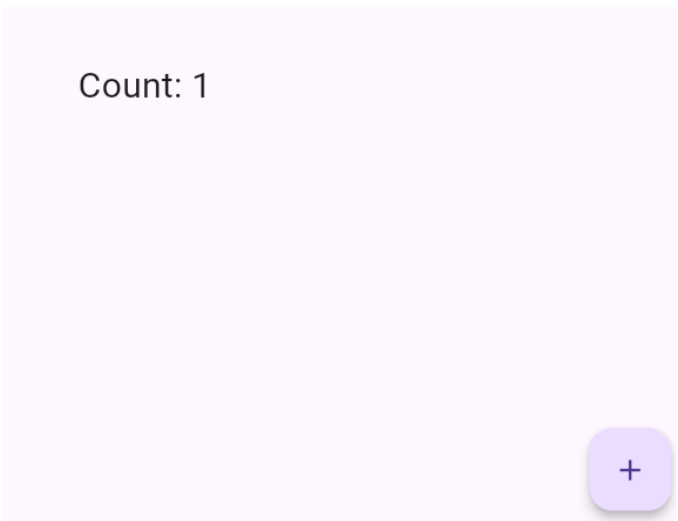
b) Implement stateful widget

Code:

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

void main() => runApp(const MaterialApp(home: CounterApp()));

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

  @override
  State<CounterApp> createState() => _CounterAppState();
}
```

```
class _CounterAppState extends State<CounterApp> {
  int count = 0;

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      body: Center(child: Text("Count: $count", style: const TextStyle(fontSize:
24))),
      floatingActionButton: FloatingActionButton(
        onPressed: () => setState(() => count++),
        child: const Icon(Icons.add),
      ),
    );
  }
}
```

Output:



Click on this then the count value increases