Tween Animation Page

import 'package:flutter/material.dart';

void main() {

runApp(MaterialApp(

debugShowCheckedModeBanner: false,

home: TweenAnimationPage(),

));

}

class TweenAnimationPage extends StatefulWidget {

@override

\_TweenAnimationPageState createState() => \_TweenAnimationPageState();

}

class \_TweenAnimationPageState extends State<TweenAnimationPage>

with SingleTickerProviderStateMixin {

late AnimationController \_controller;

late Animation<double> \_animation;

@override

void initState() {

super.initState();

\_controller =

AnimationController(vsync: this, duration: Duration(seconds: 2))

..repeat(reverse: true);

\_animation = Tween<double>(begin: 50, end: 200).animate(CurvedAnimation(

parent: \_controller,

curve: Curves.easeInOut,

));

}

@override

void dispose() {

\_controller.dispose();

super.dispose();

}

@override

Widget build(BuildContext context) {

return Scaffold(

backgroundColor: Color(0xFFFFF6FF),

appBar: AppBar(title: Text("Tween Animation Page")),

body: Center(

child: AnimatedBuilder(

animation: \_animation,

builder: (context, child) {

return FlutterLogo(size: \_animation.value);

},

),

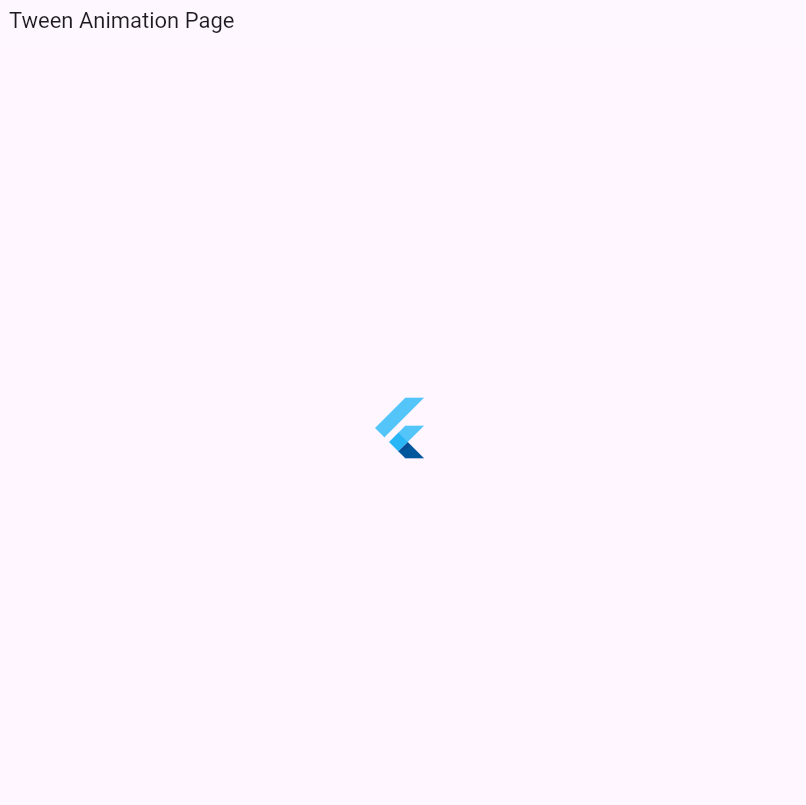
),

);

}

}

Output:



2.Diamond Shape Animation Page

import 'package:flutter/material.dart';

import 'dart:math' as math;

void main() {

runApp(MaterialApp(

debugShowCheckedModeBanner: false,

home: DiamondShapeAnimationPage(),

));

}

class DiamondShapeAnimationPage extends StatefulWidget {

@override

\_DiamondShapeAnimationPageState createState() =>

\_DiamondShapeAnimationPageState();

}

class \_DiamondShapeAnimationPageState extends State<DiamondShapeAnimationPage>

with SingleTickerProviderStateMixin {

late AnimationController \_controller;

late Animation<double> \_rotationAnimation;

@override

void initState() {

super.initState();

\_controller =

AnimationController(vsync: this, duration: Duration(seconds: 2))

..repeat(reverse: true);

\_rotationAnimation =

Tween<double>(begin: 0, end: math.pi / 4).animate(CurvedAnimation(

parent: \_controller,

curve: Curves.easeInOut,

));

}

@override

void dispose() {

\_controller.dispose();

super.dispose();

}

@override

Widget build(BuildContext context) {

return Scaffold(

backgroundColor: Color(0xFFFFF6FF),

appBar: AppBar(title: Text("Diamond Shape Animation")),

body: Center(

child: AnimatedBuilder(

animation: \_rotationAnimation,

builder: (context, child) {

return Transform.rotate(

angle: \_rotationAnimation.value,

child: Container(

width: 100,

height: 100,

color: Colors.blue,

),

);

},

),

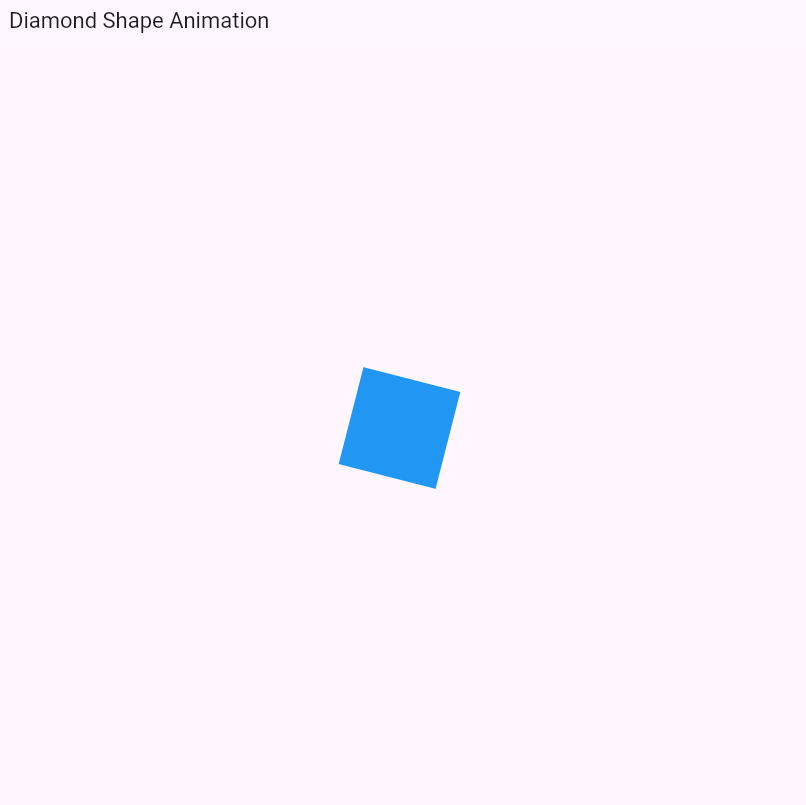
),

);

}

}

OUTPUT:



DRAG PHYSICS:

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: 'Drag Physics Example',

theme: ThemeData(

colorScheme: ColorScheme.fromSeed(seedColor: Colors.deepPurple),

useMaterial3: true,

),

home: const DragPhysicsExample(),

);

}

}

class DragPhysicsExample extends StatefulWidget {

const DragPhysicsExample({super.key});

@override

\_DragPhysicsExampleState createState() => \_DragPhysicsExampleState();

}

class \_DragPhysicsExampleState extends State<DragPhysicsExample> {

Offset \_offset = const Offset(100, 100);

@override

Widget build(BuildContext context) {

return Scaffold(

appBar: AppBar(

title: const Text('Drag Physics Example'),

),

body: Stack(

children: <Widget>[

Positioned(

left: \_offset.dx,

top: \_offset.dy,

child: GestureDetector(

onPanUpdate: (details) {

setState(() {

\_offset += details.delta;

});

},

child: Container(

width: 100,

height: 50,

color: Colors.blue,

alignment: Alignment.center,

child: const Text(

'Drag Me!',

style: TextStyle(color: Colors.white),

),

),

),

),

],

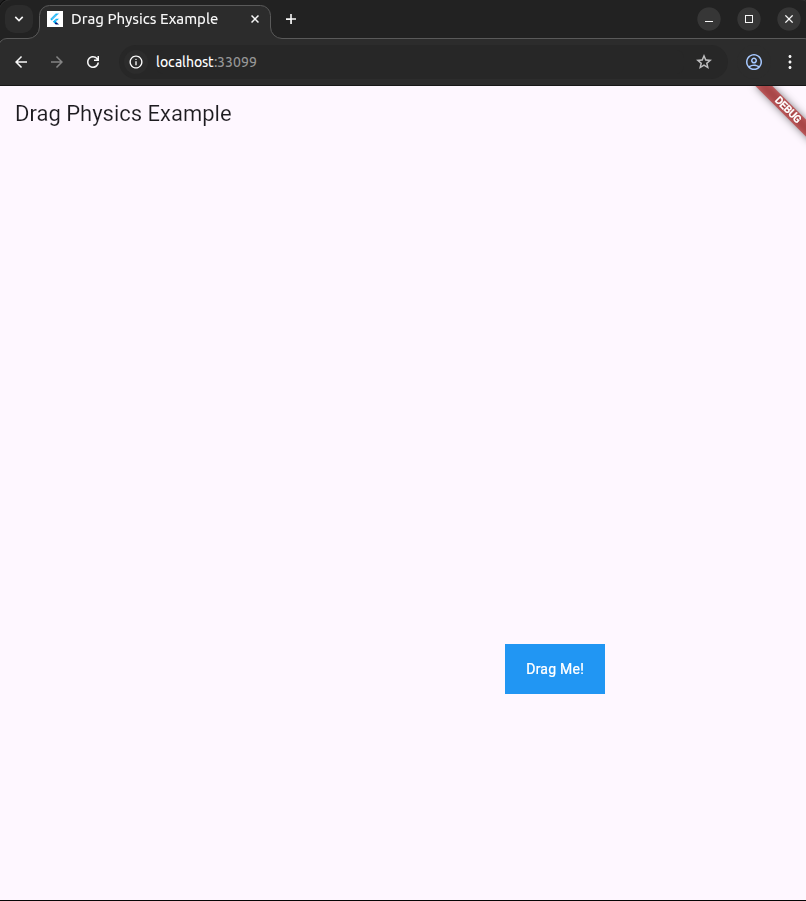
),

);

}

}

OUTPUT:



OUT OF BOX:

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: 'Out of Box Animation',

theme: ThemeData(

colorScheme: ColorScheme.fromSeed(seedColor: Colors.deepPurple),

useMaterial3: true,

),

home: const OutOfBoxAnimation(),

);

}

}

class OutOfBoxAnimation extends StatefulWidget {

const OutOfBoxAnimation({super.key});

@override

State<OutOfBoxAnimation> createState() => \_OutOfBoxAnimationState();

}

class \_OutOfBoxAnimationState extends State<OutOfBoxAnimation> {

double \_size = 100.0;

bool \_isEnlarged = false;

void \_animateBox() {

setState(() {

\_size = \_isEnlarged ? 100.0 : 200.0;

\_isEnlarged = !\_isEnlarged;

});

}

@override

Widget build(BuildContext context) {

return Scaffold(

appBar: AppBar(

title: const Text('Out of Box Animation'),

),

body: Center(

child: GestureDetector(

onTap: \_animateBox,

child: AnimatedContainer(

duration: const Duration(milliseconds: 500),

curve: Curves.fastOutSlowIn,

width: \_size,

height: \_size,

color: Colors.blue,

alignment: Alignment.center,

child: const Text(

'Box',

style: TextStyle(color: Colors.white),

),

),

),

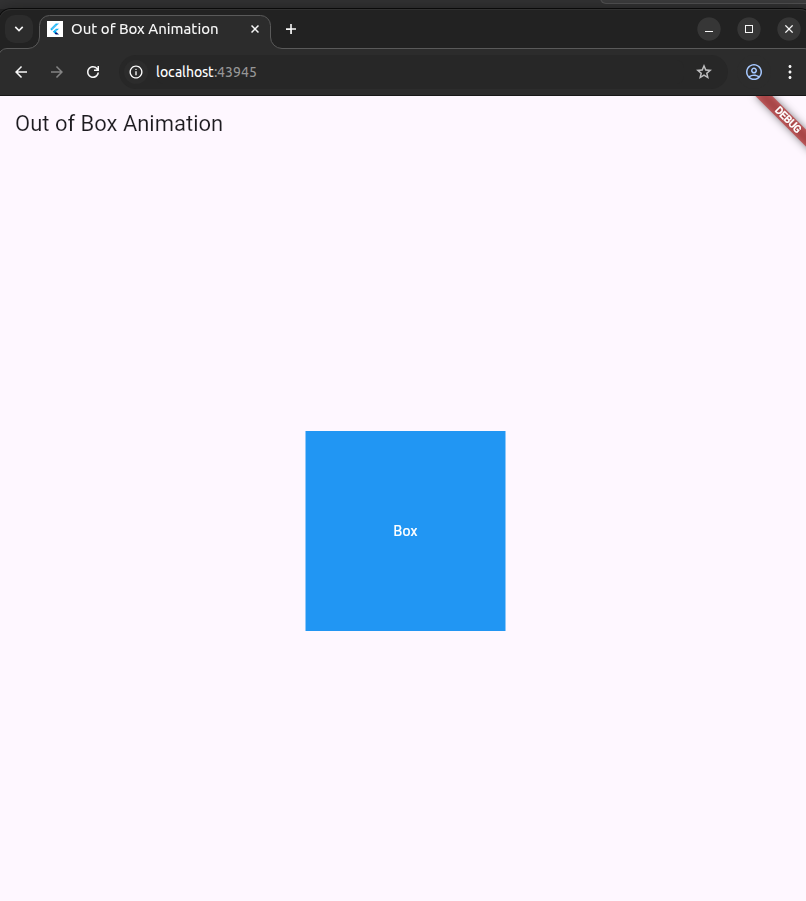
),

);

}

}

OUTPUT:



ANIMATED CONTAINER:

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: 'AnimatedContainer Sample',

theme: ThemeData(

colorScheme: ColorScheme.fromSeed(seedColor: Colors.deepPurple),

useMaterial3: true,

),

home: const AnimatedContainerPage(),

);

}

}

class AnimatedContainerPage extends StatefulWidget {

const AnimatedContainerPage({super.key});

@override

\_AnimatedContainerPageState createState() => \_AnimatedContainerPageState();

}

class \_AnimatedContainerPageState extends State<AnimatedContainerPage> {

double \_height = 100.0;

double \_width = 100.0;

Color \_color = Colors.blue;

void \_changeContainerProperties() {

setState(() {

\_height = \_height == 100.0 ? 200.0 : 100.0;

\_width = \_width == 100.0 ? 200.0 : 100.0;

\_color = \_color == Colors.blue ? Colors.red : Colors.blue;

});

}

@override

Widget build(BuildContext context) {

return Scaffold(

appBar: AppBar(

title: const Text('AnimatedContainer Sample'),

),

body: Center(

child: GestureDetector(

onTap: \_changeContainerProperties,

child: AnimatedContainer(

duration: const Duration(seconds: 1),

curve: Curves.fastOutSlowIn,

height: \_height,

width: \_width,

color: \_color,

child: const Center(

child: FlutterLogo(

size: 50,

),

),

),

),

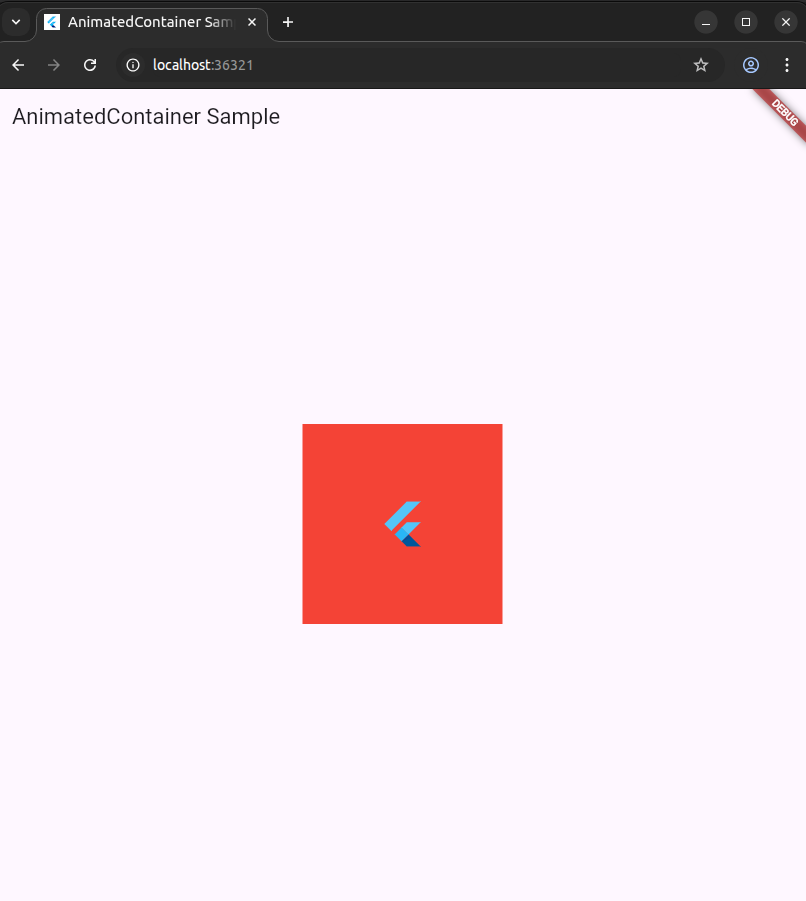
),

);

}

}

OUTPUT:



ANIMATED CROSSFADE:

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: 'AnimatedCrossFade Example',

theme: ThemeData(

colorScheme: ColorScheme.fromSeed(seedColor: Colors.deepPurple),

useMaterial3: true,

),

home: const AnimatedCrossFadePage(),

);

}

}

class AnimatedCrossFadePage extends StatefulWidget {

const AnimatedCrossFadePage({super.key});

@override

State<AnimatedCrossFadePage> createState() => \_AnimatedCrossFadePageState();

}

class \_AnimatedCrossFadePageState extends State<AnimatedCrossFadePage> {

bool \_showFirst = true;

void \_toggleCrossFade() {

setState(() {

\_showFirst = !\_showFirst;

});

}

@override

Widget build(BuildContext context) {

return Scaffold(

appBar: AppBar(

title: const Text('AnimatedCrossFade Example'),

),

body: Center(

child: Column(

mainAxisAlignment: MainAxisAlignment.center,

children: <Widget>[

AnimatedCrossFade(

duration: const Duration(seconds: 1),

crossFadeState: \_showFirst ? CrossFadeState.showFirst : CrossFadeState.showSecond,

firstChild: Container(

width: 200,

height: 150,

color: Colors.blue,

alignment: Alignment.center,

child: const Text(

'Hello, Flutter!',

style: TextStyle(color: Colors.white, fontSize: 18),

),

),

secondChild: Container(

width: 200,

height: 150,

color: Colors.red,

alignment: Alignment.center,

child: const Text(

'Goodbye, Flutter!',

style: TextStyle(color: Colors.white, fontSize: 18),

),

),

),

const SizedBox(height: 20),

ElevatedButton(

onPressed: \_toggleCrossFade,

child: const Text('Toggle'),

),

],

),

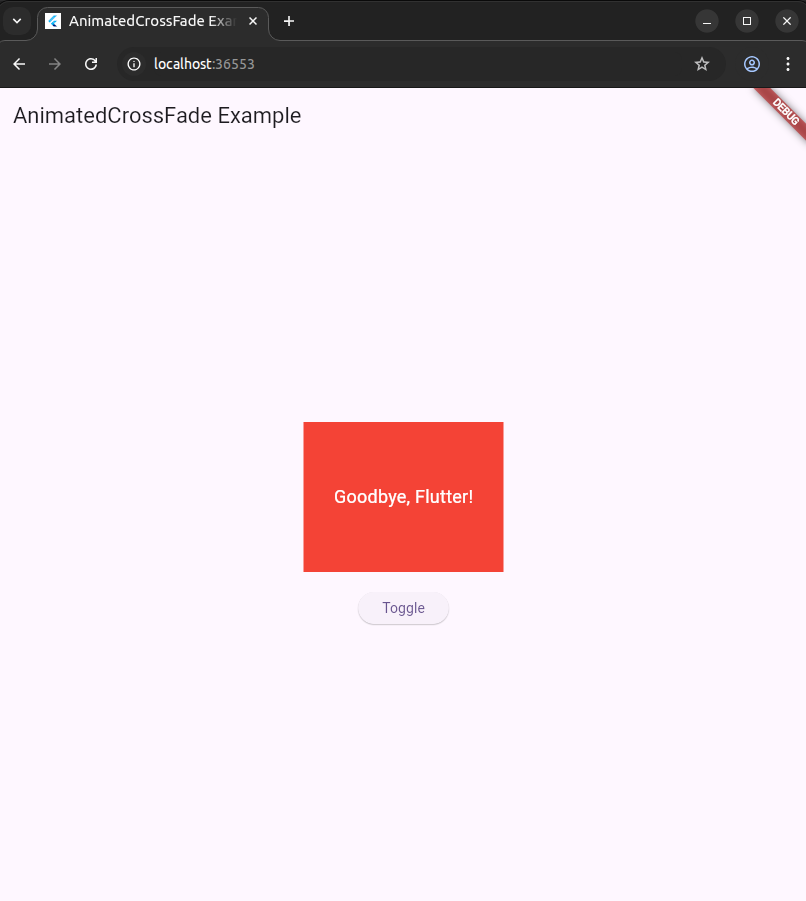
),

);

}

}

OUTPUT:



ANIMATION LIST:

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: 'AnimatedList Demo',

theme: ThemeData(

colorScheme: ColorScheme.fromSeed(seedColor: Colors.deepPurple),

useMaterial3: true,

),

home: const AnimatedListPage(),

);

}

}

class AnimatedListPage extends StatefulWidget {

const AnimatedListPage({super.key});

@override

State<AnimatedListPage> createState() => \_AnimatedListPageState();

}

class \_AnimatedListPageState extends State<AnimatedListPage> {

final GlobalKey<AnimatedListState> \_listKey = GlobalKey<AnimatedListState>();

final List<String> \_data = [

'Item 0',

'Item 1',

'Item 3',

'Item 4',

'Item 5',

'Item 2'

];

@override

Widget build(BuildContext context) {

return Scaffold(

appBar: AppBar(

title: const Text('AnimatedList'),

actions: [

IconButton(

icon: const Icon(Icons.add),

onPressed: () {

// This is a placeholder for adding an item

// For a real-world app, you would insert an item here.

},

),

],

),

body: AnimatedList(

key: \_listKey,

initialItemCount: \_data.length,

itemBuilder: (context, index, animation) {

return SlideTransition(

position: animation.drive(

Tween<Offset>(

begin: const Offset(1, 0),

end: const Offset(0, 0),

),

),

child: \_buildItem(\_data[index], index),

);

},

),

);

}

Widget \_buildItem(String item, int index) {

// A list of colors to cycle through for the items

final List<Color> colors = [

Colors.red,

Colors.pink,

Colors.deepPurple,

Colors.indigo,

Colors.blue,

Colors.green,

];

return Container(

height: 70,

margin: const EdgeInsets.symmetric(horizontal: 16, vertical: 8),

decoration: BoxDecoration(

color: colors[index % colors.length],

borderRadius: BorderRadius.circular(8),

),

alignment: Alignment.center,

child: Text(

item,

style: const TextStyle(

color: Colors.white,

fontSize: 20,

fontWeight: FontWeight.bold,

),

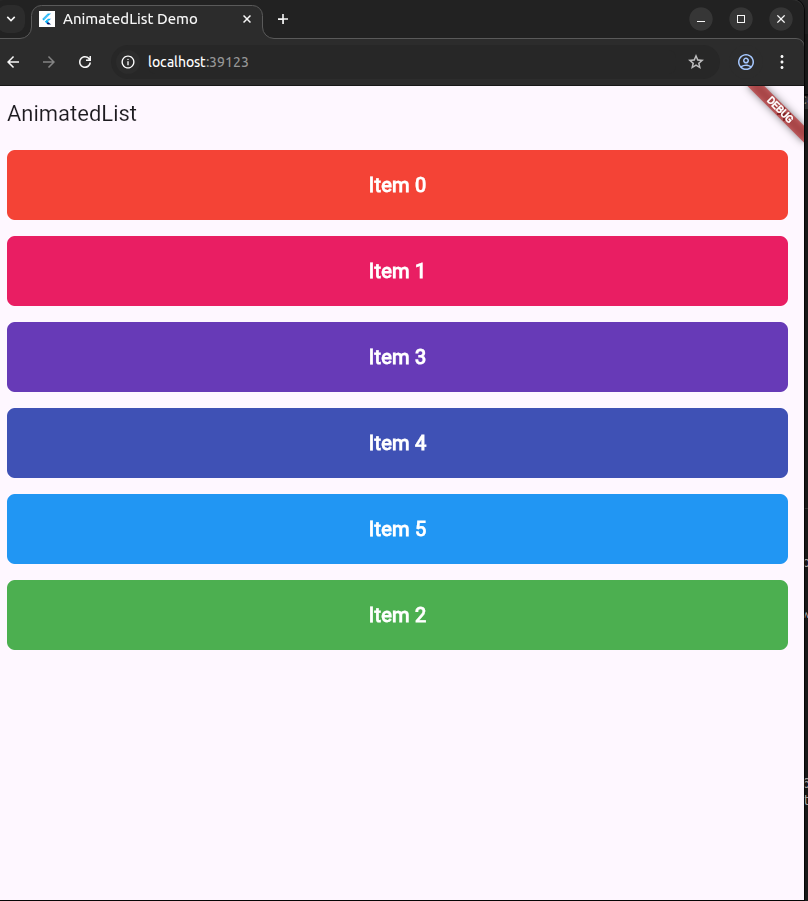
),

);

}

}

OUTPUT:



HERO ANIMATION:

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: 'Hero Animation Demo',

theme: ThemeData(

colorScheme: ColorScheme.fromSeed(seedColor: Colors.deepPurple),

useMaterial3: true,

),

home: const FirstPage(),

);

}

}

class FirstPage extends StatelessWidget {

const FirstPage({super.key});

@override

Widget build(BuildContext context) {

return Scaffold(

appBar: AppBar(

title: const Text('Hero Sample'),

),

body: Center(

child: Column(

mainAxisAlignment: MainAxisAlignment.center,

children: [

GestureDetector(

onTap: () {

Navigator.of(context).push(

MaterialPageRoute(builder: (context) => const SecondPage()),

);

},

child: Hero(

tag: 'hero-square',

child: Container(

width: 50,

height: 50,

color: Colors.blue,

),

),

),

const SizedBox(height: 20),

const Text('Tap on the icon to view hero animation transition.'),

],

),

),

);

}

}

class SecondPage extends StatelessWidget {

const SecondPage({super.key});

@override

Widget build(BuildContext context) {

return Scaffold(

appBar: AppBar(

title: const Text('Second Page'),

),

body: Center(

child: Hero(

tag: 'hero-square',

child: Container(

width: 200,

height: 200,

color: Colors.blue,

),

),

),

);

}

}

OUTPUT:

