|  |  |
| --- | --- |
| **Experiment** | 6 |
| **Aim** | Create an app for children where children can learn numbers and alphabets |
| **Objective** | * To Create App of children * To use Upper Tabs in navigation bar * To implement Splash Screen in Flutter |
| **Name** | Atharva Angre |
| **UCID** | 2024510001 |
| **Class** | FYMCA |
| **Batch** | A |
| **Date of Submission** | 09.03.25 |

|  |  |
| --- | --- |
| **Technology used** | **Flutter, Android Studio** |
| **Task** | Children should be able to even recognize the numbers and alphabets by quiz form.  Create 4 upper Tabs with Splash Screen at the start:  1) Practice/Lessons  2) Quiz  3) Leader Board  4) Profile |
| **Code with proper label** | Main.dart import 'package:flutter/material.dart'; import 'package:get/get.dart';  import 'Screens/splashScreen.dart';   void main() {  runApp(MyApp()); }  class MyApp extends StatelessWidget {  @override  Widget build(BuildContext context) {  return GetMaterialApp(  debugShowCheckedModeBanner: false,  title: 'Kids Learning App',  theme: ThemeData(  primaryColor: Color(0xFF4A90E2),  colorScheme: ColorScheme.fromSwatch().copyWith(  secondary: Color(0xFFFFA726),  background: Color(0xFFF5F5F5)  ),  fontFamily: 'ComicSans'  ),  home: SplashScreen(),  );  } }  splashScreen.dart import 'package:flutter/material.dart'; import 'package:get/get.dart';  import 'homeScreen.dart';  class SplashScreen extends StatefulWidget {  const SplashScreen({super.key});   @override  State<SplashScreen> createState() => \_SplashScreenState(); }  class \_SplashScreenState extends State<SplashScreen>  with SingleTickerProviderStateMixin {  late AnimationController \_controller;  late Animation<double> \_scaleAnimation;   @override  void initState() {  super.initState();   // Initialize animation controller  \_controller = AnimationController(  duration: const Duration(seconds: 2),  vsync: this,  );   // Scale animation from 0.5x to 1.0x  \_scaleAnimation = Tween<double>(begin: 0.5, end: 1.0).animate(  CurvedAnimation(parent: \_controller, curve: Curves.*easeInOut*),  );   // Start animation  \_controller.forward();   // Navigate to HomeScreen after animation completes  Future.delayed(const Duration(seconds: 3), () {  Get.off(() => const HomeScreen());  });  }   @override  void dispose() {  \_controller.dispose();  super.dispose();  }   @override  Widget build(BuildContext context) {  return Scaffold(  body: Center(  child: ScaleTransition(  scale: \_scaleAnimation,  child: Hero(  tag: 'SPIT\_LOGO',  child: ClipRRect(  borderRadius: BorderRadius.circular(8), // Ensure same shape  child: Image.asset(  "assets/images/spit.jpeg",  width: 150,  height: 150,  fit: BoxFit.cover,  ),  ),  ),  ),  ),  );  } }  homeScreen.dart import 'package:flutter/material.dart'; import 'package:get/get.dart'; import 'practice\_screen.dart'; import 'quiz\_screen.dart'; import 'leaderboard\_screen.dart'; import 'profile\_screen.dart';  class HomeScreen extends StatefulWidget {  const HomeScreen({super.key});   @override  State<HomeScreen> createState() => \_HomeScreenState(); }  class \_HomeScreenState extends State<HomeScreen> {  @override  Widget build(BuildContext context) {  return DefaultTabController(  length: 4,  child: Scaffold(  appBar: AppBar(  backgroundColor: Color(0xFF4A90E2),  title: const Text('Kids Learning App', style: TextStyle(color: Colors.*white*, fontSize: 20)),  bottom: const TabBar(  labelColor: Colors.*white*,  indicatorColor: Colors.*orange*,  tabs: [  Tab(icon: Icon(Icons.*school*), text: 'Practice'),  Tab(icon: Icon(Icons.*quiz*), text: 'Quiz'),  Tab(icon: Icon(Icons.*leaderboard*), text: 'Leaderboard'),  Tab(icon: Icon(Icons.*person*), text: 'Profile'),  ],  ),  ),  body: const TabBarView(  children: [  PracticeScreen(),  QuizScreen(),  LeaderboardScreen(),  ProfileScreen(),  ],  ),  ),  );  } }  practiceScreen.dart import 'package:flutter/material.dart';  class PracticeScreen extends StatefulWidget {  const PracticeScreen({super.key});   @override  \_PracticeScreenState createState() => \_PracticeScreenState(); }  class \_PracticeScreenState extends State<PracticeScreen> {  List<String> \_numbers = List.generate(11, (index) => index.toString());  List<String> \_alphabets = List.generate(26, (index) => String.fromCharCode(65 + index));  List<String> \_currentList = [];  String \_selectedCategory = '';   void \_loadNumbers() {  setState(() {  \_currentList = \_numbers;  \_selectedCategory = 'Numbers';  });  }   void \_loadAlphabets() {  setState(() {  \_currentList = \_alphabets;  \_selectedCategory = 'Alphabets';  });  }   @override  Widget build(BuildContext context) {  return Scaffold(  backgroundColor: Colors.*blue*.shade100,  body: Center(  child: Padding(  padding: const EdgeInsets.all(20.0),  child: Column(  mainAxisAlignment: MainAxisAlignment.center,  children: [  if (\_selectedCategory.isEmpty) ...[  Text(  'Practice Numbers & Alphabets',  style: TextStyle(fontSize: 22, fontWeight: FontWeight.*bold*, color: Colors.*black87*),  textAlign: TextAlign.center,  ),  SizedBox(height: 20),  ElevatedButton(  onPressed: \_loadNumbers,  style: ElevatedButton.*styleFrom*(  backgroundColor: Colors.*orange*,  padding: EdgeInsets.symmetric(horizontal: 30, vertical: 15),  shape: RoundedRectangleBorder(borderRadius: BorderRadius.circular(10)),  ),  child: Text('Practice Numbers', style: TextStyle(fontSize: 18, color: Colors.*white*)),  ),  SizedBox(height: 10),  ElevatedButton(  onPressed: \_loadAlphabets,  style: ElevatedButton.*styleFrom*(  backgroundColor: Colors.*orange*,  padding: EdgeInsets.symmetric(horizontal: 30, vertical: 15),  shape: RoundedRectangleBorder(borderRadius: BorderRadius.circular(10)),  ),  child: Text('Practice Alphabets', style: TextStyle(fontSize: 18, color: Colors.*white*)),  ),  ] else ...[  Text(  'Practicing $\_selectedCategory',  style: TextStyle(fontSize: 22, fontWeight: FontWeight.*bold*, color: Colors.*black87*),  textAlign: TextAlign.center,  ),  SizedBox(height: 20),  Expanded(  child: GridView.builder(  gridDelegate: SliverGridDelegateWithFixedCrossAxisCount(  crossAxisCount: 3,  crossAxisSpacing: 10,  mainAxisSpacing: 10,  ),  itemCount: \_currentList.length,  itemBuilder: (context, index) {  return Container(  decoration: BoxDecoration(  color: Colors.*orange*,  borderRadius: BorderRadius.circular(10),  ),  child: Center(  child: Text(  \_currentList[index],  style: TextStyle(fontSize: 32, fontWeight: FontWeight.*bold*, color: Colors.*white*),  ),  ),  );  },  ),  ),  SizedBox(height: 20),  ElevatedButton(  onPressed: () {  setState(() {  \_selectedCategory = '';  \_currentList = [];  });  },  style: ElevatedButton.*styleFrom*(  backgroundColor: Colors.*redAccent*,  padding: EdgeInsets.symmetric(horizontal: 30, vertical: 15),  shape: RoundedRectangleBorder(borderRadius: BorderRadius.circular(10)),  ),  child: Text('Back', style: TextStyle(fontSize: 18, color: Colors.*white*)),  ),  ],  ],  ),  ),  ),  );  } }  quizScreen.dart import 'package:flutter/material.dart'; import 'package:lottie/lottie.dart';  class QuizScreen extends StatefulWidget {  const QuizScreen({super.key});   @override  \_QuizScreenState createState() => \_QuizScreenState(); }  class \_QuizScreenState extends State<QuizScreen> {  int \_currentQuestionIndex = 0;  int \_score = 0;  List<Map<String, dynamic>> \_questions = [  {  'question': 'What is the number after 2?',  'options': ['1', '2', '3', '4'],  'answer': '3'  },  {  'question': 'Which letter comes after A?',  'options': ['C', 'B', 'D', 'E'],  'answer': 'B'  },  {  'question': 'How many legs does a cat have?',  'options': ['2', '3', '4', '5'],  'answer': '4'  },  {  'question': 'What color is the sky?',  'options': ['Green', 'Blue', 'Red', 'Yellow'],  'answer': 'Blue'  },  {  'question': 'Which fruit is yellow and curved?',  'options': ['Apple', 'Banana', 'Grapes', 'Strawberry'],  'answer': 'Banana'  }  ];   void \_checkAnswer(String selectedOption) {  if (selectedOption == \_questions[\_currentQuestionIndex]['answer']) {  \_score++;  }  \_nextQuestion();  }   void \_nextQuestion() {  setState(() {  if (\_currentQuestionIndex < \_questions.length - 1) {  \_currentQuestionIndex++;  } else {  \_showResult();  }  });  }   void \_showResult() {  showDialog(  context: context,  barrierDismissible: false,  builder: (context) => AlertDialog(  title: Text("Quiz Completed!"),  content: Column(  mainAxisSize: MainAxisSize.min,  children: [  Lottie.*asset*('assets/success.json', width: 200, height: 200),  SizedBox(height: 10),  Text("Your Score: $\_score / ${\_questions.length}", style: TextStyle(fontSize: 18, fontWeight: FontWeight.*bold*)),  ],  ),  actions: [  TextButton(  onPressed: () {  setState(() {  \_currentQuestionIndex = 0;  \_score = 0;  });  Navigator.*pop*(context);  },  child: Text("Play Again"),  )  ],  ),  );  }   @override  Widget build(BuildContext context) {  return Scaffold(  backgroundColor: Colors.*blue*.shade100,  body: Padding(  padding: const EdgeInsets.all(20.0),  child: Column(  mainAxisAlignment: MainAxisAlignment.center,  crossAxisAlignment: CrossAxisAlignment.center,  children: [  Text(  \_questions[\_currentQuestionIndex]['question'],  textAlign: TextAlign.center,  style: TextStyle(fontSize: 24, fontWeight: FontWeight.*bold*, color: Colors.*black87*),  ),  SizedBox(height: 20),  GridView.builder(  shrinkWrap: true,  gridDelegate: SliverGridDelegateWithFixedCrossAxisCount(  crossAxisCount: 2,  childAspectRatio: 2.5,  crossAxisSpacing: 10,  mainAxisSpacing: 10,  ),  itemCount: \_questions[\_currentQuestionIndex]['options'].length,  itemBuilder: (context, index) {  String option = \_questions[\_currentQuestionIndex]['options'][index];  return ElevatedButton(  onPressed: () => \_checkAnswer(option),  style: ElevatedButton.*styleFrom*(  backgroundColor: Colors.*orange*,  padding: EdgeInsets.symmetric(horizontal: 30, vertical: 15),  shape: RoundedRectangleBorder(borderRadius: BorderRadius.circular(10)),  ),  child: Text(  option,  style: TextStyle(fontSize: 18, color: Colors.*white*),  ),  );  },  ),  ],  ),  ),  );  } }  leadboardScreend.dart import 'package:flutter/material.dart';  class LeaderboardScreen extends StatelessWidget {  const LeaderboardScreen({super.key});   final List<Map<String, dynamic>> \_leaderboard = const [  {'name': 'Atharva', 'score': 100},  {'name': 'Abhishek', 'score': 90},  {'name': 'Abhijeet', 'score': 85},  {'name': 'Adam', 'score': 80},  {'name': 'Vineet', 'score': 75},  {'name': 'Darshan', 'score': 70},  {'name': 'Kiran', 'score': 65},  {'name': 'Ram', 'score': 60},  ];   @override  Widget build(BuildContext context) {  return Scaffold(  backgroundColor: Colors.*blue*.shade100,  appBar: AppBar(  title: Text('Leaderboard'),  backgroundColor: Color(0xFFFFA726),  centerTitle: true,  ),  body: Padding(  padding: const EdgeInsets.all(20.0),  child: Column(  children: [  Text(  'Top Performers',  style: TextStyle(fontSize: 22, fontWeight: FontWeight.*bold*, color: Colors.*black87*),  ),  SizedBox(height: 10),  Expanded(  child: ListView.builder(  itemCount: \_leaderboard.length,  itemBuilder: (context, index) {  return Card(  elevation: 3,  margin: EdgeInsets.symmetric(vertical: 8),  shape: RoundedRectangleBorder(borderRadius: BorderRadius.circular(10)),  child: ListTile(  leading: CircleAvatar(  backgroundColor: Colors.*orange*,  child: Text(  '${index + 1}',  style: TextStyle(color: Colors.*white*, fontWeight: FontWeight.*bold*),  ),  ),  title: Text(  \_leaderboard[index]['name'],  style: TextStyle(fontSize: 18, fontWeight: FontWeight.*bold*),  ),  trailing: Text(  '${\_leaderboard[index]['score']} pts',  style: TextStyle(fontSize: 16, fontWeight: FontWeight.*w500*, color: Colors.*black54*),  ),  ),  );  },  ),  ),  ],  ),  ),  );  } }  profileScreen.dart import 'package:flutter/material.dart';  class ProfileScreen extends StatelessWidget {  const ProfileScreen({super.key});   @override  Widget build(BuildContext context) {  return Scaffold(  backgroundColor: Colors.*blue*.shade100,  body: Center(  child: Padding(  padding: const EdgeInsets.all(20.0),  child: Column(  mainAxisAlignment: MainAxisAlignment.center,  crossAxisAlignment: CrossAxisAlignment.center,  children: [  CircleAvatar(  radius: 60,  backgroundImage: AssetImage('assets/images/profileImage.png'),  ),  SizedBox(height: 20),  Text(  'Atharva Angre',  style: TextStyle(fontSize: 22, fontWeight: FontWeight.*bold*, color: Colors.*black87*),  ),  SizedBox(height: 10),  Text(  'angreatharva08@example.com',  style: TextStyle(fontSize: 16, color: Colors.*grey*[700]),  ),  SizedBox(height: 30),  ElevatedButton(  onPressed: () {},  style: ElevatedButton.*styleFrom*(  backgroundColor: Colors.*orange*,  padding: EdgeInsets.symmetric(horizontal: 40, vertical: 12),  shape: RoundedRectangleBorder(borderRadius: BorderRadius.circular(10)),  ),  child: Text(  'Edit Profile',  style: TextStyle(fontSize: 18, color: Colors.*white*),  ),  ),  ],  ),  ),  ),  );  } } |

|  |  |
| --- | --- |
| **Screenshots** |  |
| **Question and Answers** | Answer the following Questions:   1. How to create Upper Tabs in Flutter?   Upper tabs in Flutter can be created using the TabBar widget inside an AppBar, along with a TabBarView wrapped in a DefaultTabController to handle switching between tab content.   1. How did you use 60-30-10 rule in your application?   The 60-30-10 rule was applied by using a primary color (60%) for the background, a secondary color (30%) for interactive elements like buttons, and an accent color (10%) for highlights or text emphasis.   1. Which new elements did you use for creating UI components?   New UI elements include GridView for option selection, Lottie animations for feedback, and AlertDialog for displaying quiz results in a more engaging way.   1. In pubspec.yaml file, what dependencies need to be there?   get for state management  lottie for animations   1. What is the use of Splash Screen?   A Splash Screen is used to show a loading screen when the app starts, improving user experience by displaying branding, animations, or preloading essential data before navigating to the main screen. |
| **Conclusion** | Through this session, I learned how to implement upper tabs using TabBar and TabBarView, effectively structure a quiz and practice module, and design a leaderboard with an engaging UI. I also understood the importance of the 60-30-10 rule in UI design, explored new UI elements like GridView and Lottie, and identified key dependencies required in pubspec.yaml. Additionally, I reinforced my understanding of the Splash Screen and its role in enhancing user experience. |