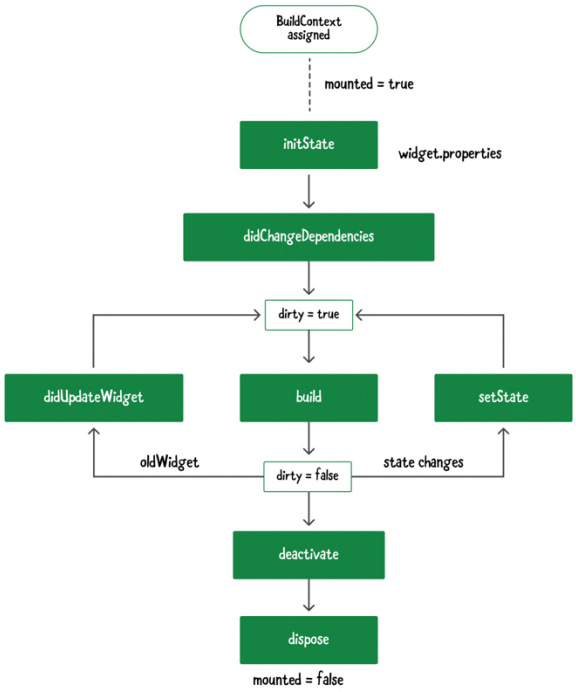
Unit-1

1.	What is cross platform mobile application development? Discuss any four advantages of cross platform mobile application development over native mobile application development.
2.	List top two mobile operating systems having major market share. Discuss any four challenges in cross platform mobile application development over native mobile application development.
3.	What is flutter? Explain four major features of flutter.
4.	Draw the flutter architecture and explain three components of it.
5.	Step-by-step explain flutter SDK installation on Windows or Mac operating system.
6.	How to create new flutter project using VS Code, Android Studio, and terminal? Also explain how to rename class name from Android Studio and VS Code.
7.	What is widget? Discuss when not to use flutter.
8.	Write sample code snippet and explain use of Scaffold, AppBar, SafeArea, and Container.
9.	How to create and add a new dart file to a flutter project in VS Code? Write Dart code snippet for creating “Employee” class with at least three properties and a constructor.
10.	Mr. Mahesh is a flutter developer. He has developed a flutter app with some deprecated Dart code. Discuss how Mr. Mahesh can find out deprecated code and ONLY correct deprecated code. Suggest the recommended alternative Dart code for deprecated “accentColor”.
11.	How to add images of dog, cat, cow to a flutter project in three separate folders? Explain step-by-step along with required configuration code.
12.	Design a Dart class with Card, Column, Image, and Text widgets. Write required code snippet and explain your solution.
13.	Analyze the following code snippet and explain each line of it: `return Scaffold(` `  appBar: AppBar(` `  title: Text(widget.recipe.label),` `  ), // AppBar` `  body: SafeArea(` `  child: Column(` `  children: <Widget>[` `  SizedBox(` `  height: 300,` `  width: double.infinity,` `  child: Image(` `  image: AssetImage(widget.recipe.imageUrl),` `  ), // Image` `  ), // SizedBox` `  const SizedBox(` `  height: 4,` `  ), // SizedBox` `  Text(` `  widget.recipe.label,` `  style: const TextStyle(fontSize: 18),` `  ), // Text`

14.	Analyze the following code snippet and explain each line of it: `Expanded(` `  child: ListView.builder(` `  padding: const EdgeInsets.all(7.0),` `  itemCount: widget.recipe.ingredients.length,` `  itemBuilder: (BuildContext context, int index) {` `  final ingredient = widget.recipe.ingredients[index];` `  return Text('${ingredient.quantity * _sliderVal} '` `  '${ingredient.measure} '` `  '${ingredient.name}'); // Text` `  },` `  ), // ListView.builder` `), // Expanded`
15.	Analyze the following code snippet and explain each line of it: `Slider(` `  min: 1,` `  max: 10,` `  divisions: 9,` `  label: '${_sliderVal * widget.recipe.servings} servings',` `  value: _sliderVal.toDouble(),` `  onChanged: (newValue) {` `  setState(() {` `  _sliderVal = newValue.round();` `  });` `  },` `  // 14` `  activeColor: Colors.green,` `  inactiveColor: Colors.black,` `), // Slider`
16.	What is the use of pubspec.yaml file? Explain at least four information from it.
17.	
18.	
19.	
20.	

Unit-2

1.	Differentiate “runApp( )” and “main( )” functions. Also explain, when to use material.dart package.
2.	List any four category of widgets and explain any two out of it.
3.	With appropriate example and required code snippet, explain input and layout widgets.
4.	List at least four material widgets and explain any one with necessary code snippet.
5.	List at least four layout widgets and explain any one with necessary code snippet.
6.	Develop a Dart class including “Padding” and “BoxDecoration” of

	"Container" widget. Discuss your code snippet.
7.	Create a flutter app demonstrating use of Row and Column widgets. Explain your solution.
8.	How to create flutter project from terminal? Write command to add dependency in a flutter project. Explain using example of google_fonts.
9.	Explain AppBar and SnackBar of Scaffold with appropriate example and code snippet.
10.	Explain BottomNavigationBar of Scaffold with appropriate example and code snippet.
11.	Create a flutter app demonstrating use of Stack and Positioned widgets. Explain your solution.
12.	Design a flutter app to display author avatar like given below:  Write necessary code snippet and explain your solution.
13.	With respect to Row widget, write at least four different values of MainAxisAlignment. Draw the diagrams and explain meaning of any two MainAxisAlignment values.
14.	Design a flutter app demonstrating use of Wrap and Chip widgets. Explain your solution.
15.	Analyze and explain the following diagram: 
16.	What is the use of ListView widget? Explain any two constructors of ListView.

17.	Analyze the following code snippet and explain each line of it: `// 1` `return FutureBuilder(` `  // 2` `  future: mockService.getExploreData(),` `  // 3` `  builder: (context, snapshot) {` `  // TODO: Add Nested List Views` `  // 4` `  if (snapshot.connectionState == ConnectionState.done) {` `  // 5` `  final recipes = snapshot.data?.todayRecipes;` `  // TODO: Replace this with TodayRecipeListView` `  return Center(` `  child:` `  Container(child: const Text('Show TodayRecipeListView')));` `  } else {` `  // 6` `  return const Center(child: CircularProgressIndicator());` `  }` `  }); // FutureBuilder`
18.	List any five key parameters of GridView and explain any two parameters.
19.	How to add google_fonts package in flutter project from terminal? Write the code it will add in pubspec.yaml.
20.	What is Scaffold? Analyze and explain following code: `import 'package:flutter/material.dart';` `// 1` `class Home extends StatefulWidget {` `  const Home({Key? key}) : super(key: key);` `  @override` `  _HomeState createState() => _HomeState();` `}` `class _HomeState extends State<Home> {` `  @override` `  Widget build(BuildContext context) {` `  return Scaffold(` `  appBar: AppBar(` `  title: Text('Fooderlich',` `  // 2` `  style: Theme.of(context).textTheme.titleLarge)), /` `  body: Center(` `  child: Text('Let\'s get cooking 👨‍🍳',` `  // 3` `  style: Theme.of(context).textTheme.displayLarge)),` `  ); // Scaffold` `  }` `}`
21.	

Unit-3

1.	Write sample code snippet and explain use of MaterialButton.
2.	What is Provider? Compare Provider with Callback.
3.	Explain major steps for adding and using Provider in a flutter project.
4.	Consider suitable example and design a Dart class extending ChangeNotifier. Discuss required code snippet.
5.	Demonstrate use of “Provider.of” with required code snippet.
6.	Mr. Manoj is learning flutter development. He is struggling with adding dependency in the flutter project. Step-by-step explain, how he can add

	dependency in his flutter project. By mistake, Mr. Manoj has added older version of dependency. How to correct this mistake? Discuss.
7.	What is the use of “intl” package? Write code snippet and explain use of it.
8.	How to add “uuid” package in your flutter project? With code snippet, exlain use of it.
9.	List different types of Chip widgets. Draw the diagram and explain any two types of Chip widgets.
10.	Your team leader has instructed you to write code for accepting birth date from the user. Which widget is useful to accomplish this requirement. Write code snippet and explain your solution.
11.	Which flutter package is useful if you want to provide color selection in your flutter project? Write code snippet and discuss.
12.	Compare GestureDetector with InkWell. Give appropriate scenarios where one preferred over other.
13.	Write code snippet and explain use of Dismissable widget.
14.	Analyze the following code snippet and explain each line of it: `// TODO 11: Present GroceryItemScreen` `// 1` `final manager = Provider.of<GroceryManager>(context, listen: false);` `// 2` `Navigator.push(` `  context,` `  // 3` `  MaterialPageRoute(` `  // 4` `  builder: (context) => GroceryItemScreen(` `  // 5` `  onCreate: (item) {` `  // 6` `  manager.addItem(item);` `  // 7` `  Navigator.pop(context);` `  },` `  ), // GroceryItemScreen` `  ), // MaterialPageRoute` `);`
15.	When to use ChoiceChip and when to use FilterChip? Write necessary code snippet and explain these two widgets.
16.	How to use “Wrap with widget” functionality provided by VS Code? Step-by-step explain with suitable example.
17.	
18.	
19.	
20.	
21.	
22.	
23.	

24.	

Unit-4

1.	What are shared preferences? Create a flutter app using pref.getBool('darkTheme') and explain your solution.
2.	When to use shared preferences? Write Dart code and justify your answer with appropriate example.
3.	Write Dart code for adding required dependency in pubspec.yaml if you want to use shared preferences. Develop flutter app which stores username in shared preferences. Discuss your solution.
4.	Mr. Mahesh is learning flutter. His teacher assigned him a task to store app usage counter in shared preferences. He needs to increment counter value whenever app is launched. Write Dart code and help Mr. Mahesh in developing this demo flutter app. Add necessary comments in your code so that Mr. Mahesh can understand Dart code easily.
5.	Analyze and explain the following code snippet: `class _MainAppState extends State<MainApp> {` `  SharedPreferences? prefs;` `  TextEditingController usernameController = TextEditingController();` `  String _username = "Guest";` `  @override` `  void initState() {` `  _loadData();` `  super.initState();` `  }` `  void _loadData() async {` `  try {` `  prefs = await SharedPreferences.getInstance();` `  } catch (e) {` `  print(e);` `  }` `  setState(() {` `  _username = prefs?.getString("username") ?? "Guest";` `  });` `  }`
6.	Define serialization. Analyze and explain following code snippet:

<table>
<tr><td></td><td>

```dart
import 'package:json_annotation/json_annotation.dart';

part 'address.g.dart';

@JsonSerializable()
class Address {
  String street, landmark, city, state;

  Address(
      {required this.street,
      required this.landmark,
      required this.city,
      required this.state});

  factory Address.fromJson(Map<String, dynamic> data) {
    return _$AddressFromJson(data);
  }

  Map<String, dynamic> toJson() => _$AddressToJson(this);
}
```

</td></tr>
<tr><td>7.</td><td>What is JSON serialization? Write and explain Dart code demonstrating use of “json_annotation” and “json_serializable”.</td></tr>
<tr><td>8.</td><td>Analyze the following code snippet and discuss the errors underlined. Write and explain the command to resolve these errors.


```dart
import 'package:json_annotation/json_annotation.dart';

part 'user.g.dart';
```

Target of URI hasn't been generated: 'user.g.dart'.

```dart
class User {
  User(this.name, this.email);

  String name;
  String email;

  factory User.fromJson(Map<String, dynamic> json) => _$UserFromJson(json);

  Map<String, dynamic> toJson() => _$UserToJson(this);
}
```

</td></tr>
<tr><td>9.</td><td>How to add json_serializable dependency in a flutter project? Explain the use of following:
• dart run build_runner build
• dart run build_runner build –delete-conflicting-outputs
• dart run build_runner watch
• dart run build_runner watch –delete-conflicting-outputs</td></tr>
<tr><td>10.</td><td>Write Dart code and discuss use of automated JSON serialization using code generation.</td></tr>
<tr><td>11.</td><td>Analyze and explain the following code snippet:</td></tr>
</table>

	``dart Future<Person> fetchPerson() async { final response = await http .get(Uri.parse('http://192.168.0.103/getStudentService.php/')); if (response.statusCode == 200) { // If the server did return a 200 OK response, // then parse the JSON. return Person.fromJson(jsonDecode(response.body)); } else { // If the server did not return a 200 OK response, // then throw an exception. throw Exception('Failed to load person'); } } ```
12.	Write and explain Dart code for a flutter application to download data from web API using http package and display on screen.
13.	What is the use of chopper package? Explain at least two advantages of Chopper compare to http package.
14.	Analyze and explain the following code snippet: ```dart import 'package:chopper/chopper.dart'; part 'student_service.chopper.dart'; @ChopperApi() abstract class StudentService extends ChopperService { static StudentService create() { final client = ChopperClient(baseUrl: Uri.parse('http://192.168.0.103/getStudentService.php/'), interceptors: [_addQuery], services: [_$StudentService()], converter: const JsonConverter(),); // ChopperClient return _$StudentService(client); } static Request _addQuery(Request req) { final params = Map<String, dynamic>.from(req.parameters); return req.copyWith(parameters: params); } @Get() Future<Response> getStudent(@Query() String rank,); } ```
15.	Write and explain Dart code to demonstrate use of chopper interceptor.
16.	
17.	
18.	
19.	

20.	

Unit-5

1.	With appropriate example and necessary code snippet, explain state management using InheritedWidget.
2.	What is state management in flutter? Explain advantage and limitation of InheritedWidget.
3.	Analyze and explain use of InheritedWidget from following code snippet: `class RecipeWidget extends InheritedWidget {` `  final Recipe recipe;` `  const RecipeWidget({super.key,` `  required this.recipe, required Widget child})` `  : super(child: child);` `  @override` `  bool updateShouldNotify(RecipeWidget oldWidget) =>` `  recipe != oldWidget.recipe;` `  static RecipeWidget? of(BuildContext context) =>` `  context.dependOnInheritedWidgetOfExactType<RecipeWidget>();` `}`
4.	Explain use of following with respect to provider: • ChangeNotifier • ChangeNotifierProvider • Consumer • FutureProvider • MultiProvider
5.	Compare InheritedWidget with Provider and discuss one advantage of each.
6.	What is the use of Consumer with respect to state management using Provider? Write Dart code snippet and explain.
7.	With appropriate example and necessary code snippet, explain use of context.watch<T>().
8.	With appropriate example and necessary code snippet, explain use of context.select<T, R>().
9.	When to use of FutureProvider? Analyze and explain following code snippet: FutureProvider(create: (context) => createFuture(), child: <widget>,);

	Future<MyModel> createFuture() async { return Future.value(MyModel()); }
10.	When to use of MultiProvider? Explain use of MultiProvider with necessary Dart code.
11.	What is ProxyProvider? When to use it? Discuss with appropriate example.
12.	What is the use of ChangeNotifier? Analyze and explain following Dart code: `class Counter extends ChangeNotifier {` `  Counter() {` `  Timer.periodic(const Duration(seconds: 1), (timer) {` `  _count++;` `  notifyListeners();` `  }); // Timer.periodic` `  }` `  int _count = 42;` `  int get count => _count;` `}`
13.	When to use context.read<T>()? Analyze the following code snippet and explain usage of context.watch<T>(). `class MyApp extends StatelessWidget {` `  const MyApp({super.key});` `  @override` `  Widget build(BuildContext context) {` `  final dateTime = context.watch<Clock>().dateTime;` `  final count = context.watch<Counter>().count;` `  return Center(` `  child: RichText(` `  text: TextSpan(` `  text: "$dateTime\n\n$count",` `  style: const TextStyle(fontSize: 36),` `  ), // TextSpan` `  textDirection: TextDirection.ltr,` `  ), // RichText` `  ); // Center` `  }` `}`
14.	What is MultiProvider? When to use MultiProvider? Analyze and explain following Dart code:

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

Run | Debug | Profile
void main() {
  runApp(MultiProvider(
    providers: [
      ChangeNotifierProvider(create: (_) => Counter()),
      ChangeNotifierProvider(create: (_) => Clock())
    ],
    child: const MyApp(),
  )); // MultiProvider
}
```

15. What is the use of ChangeNotifierProvider? Analyze and explain following Dart code:

```dart
import 'package:flutter/foundation.dart';
import '/models/catalog.dart';
class CartModel extends ChangeNotifier {
  late CatalogModel _catalog;
  final List<int> _itemIds = [];
  CatalogModel get catalog => _catalog;
  set catalog(CatalogModel newCatalog) {
    _catalog = newCatalog;
    notifyListeners();
  }
  List<Item> get items => _itemIds.map((id) => _catalog.getById(id)).toList();
  int get totalPrice =>
      items.fold(0, (total, current) => total + current.price);
  void add(Item item) {
    _itemIds.add(item.id);
    notifyListeners();
  }
  void remove(Item item) {
    _itemIds.remove(item.id);
    notifyListeners();
  }
}
```

16. What is silvers? Analyze following code snippet and explain usage of silvers.

```dart
import 'package:flutter/material.dart';
import 'package:go_router/go_router.dart';
import 'package:provider/provider.dart';
import '/models/cart.dart';
import '/models/catalog.dart';

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

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      body: CustomScrollView(
        slivers: [
          _MyAppBar(),
          const SliverToBoxAdapter(child: SizedBox(height: 12)),
          SliverList(
            delegate: SliverChildBuilderDelegate(
                (context, index) => _MyListItem(index)), // Sliv
          ), // SliverList
        ],
      ), // CustomScrollView
    ); // Scaffold
  }
}
```

17. Analyze the following Dart code and explain context.select<T, R>() and context.read<T>().

```dart
class _AddButton extends StatelessWidget {
  final Item item;
  const _AddButton({required this.item});
  @override
  Widget build(BuildContext context) {
    var isInCart = context.select<CartModel, bool>(
      (cart) => cart.items.contains(item),
    );
    return TextButton(
      onPressed: isInCart
          ? null
          : () {
              var cart = context.read<CartModel>();
              cart.add(item);
            },
      style: ButtonStyle(
        overlayColor: MaterialStateProperty.resolveWith<Color?>((states) {
          if (states.contains(MaterialState.pressed)) {
            return Theme.of(context).primaryColor;
          }
          return null; // Defer to the widget's default.
        }),
      ), // ButtonStyle
      child: isInCart
          ? const Icon(Icons.check, semanticLabel: 'ADDED')
          : const Text('ADD'),
```

18. What is the use of ChangeNotifierProxyProvider? Analyze the following code and explain:

	``dart class MainApp extends StatelessWidget { const MainApp({super.key}); @override Widget build(BuildContext context) { return MultiProvider(providers: [Provider(create: (context) => CatalogModel()), ChangeNotifierProxyProvider<CatalogModel, CartModel>(create: (context) => CartModel(), update: (context, catalog, cart) { if (cart == null) throw ArgumentError.notNull('cart'); cart.catalog = catalog; return cart; },), // ChangeNotifierProxyProvider], child: MaterialApp.router(title: 'Provider Demo', theme: appTheme, routerConfig: router(),), // MaterialApp.router); // MultiProvider } ```
19.	Draw the diagram and explain working of Redux and BLoC.
20.	What is Riverpod? Explain any two limitations of Provider which are solved by Riverpod.
21.	With diagram, explain repository design pattern. Discuss any two scenarios where repository design pattern is suitable.
22.	

Unit-6

1.	What is asynchronous function? Explain any two advantages of stream.
2.	With appropriate example, explain how to create stream with necessary Dart code snippet.
3.	What is single subscription stream? Explain it with appropriate example and Dart code.
4.	What is broadcast stream? Explain it with appropriate example and Dart code.
5.	Compare and differentiate single subscription streams and broadcast streams.
6.	With appropriate example and necessary code snippet, explain usage of StreamController, Stream, and StreamSink.
7.	What is StreamBuilder? With appropriate example and necessary code snippet, explain usage of StreamBuilder.
8.	With appropriate example and necessary code snippet, explain usage of StreamSubscription.
9.	Analyze and explain following code snippet:

```dart
class CounterApp extends StatefulWidget {
  const CounterApp({Key? key}) : super(key: key);

  @override
  _CounterAppState createState() => _CounterAppState();
}

class _CounterAppState extends State<CounterApp> {
  final StreamController _controller = StreamController();
  int _counter = 10;

  void startTimer() async {
    Timer.periodic(const Duration(seconds: 1), (timer) {
      _counter--;

      // add event/data to stream controller using sink
      _controller.sink.add(_counter);

      if (_counter <= 0) {
        timer.cancel();
      }
    }); // Timer.periodic
  }
```

10. Analyze following code snippet and explain use of StreamBuilder.

```dart
@override
void dispose() {
  super.dispose(); _controller.close();
}
@override
Widget build(BuildContext context) {
  return Scaffold(
    body: Center(
      child: Column(
        mainAxisAlignment: MainAxisAlignment.center,
        children: [
          StreamBuilder( initialData: _counter,
              stream: _controller.stream,
              builder: (context, snapshot) {
                return Text('${snapshot.data}');
              }), // StreamBuilder
          const SizedBox( height: 20, ),
          ElevatedButton(
              onPressed: () { _counter = 10;
                startTimer(); },
              child: const Text('Start Count Down'))
        ],
```

11. Analyze and explain following code snippet:

	`` ElevatedButton(onPressed: () { setState(() { result = "\n"; Stream<String> stream = broadcastController.stream; StreamSubscription<String> subscriber1 = stream.listen((String data) { result += "Subscriber1: $data\n"; }, onError: (error) { result += "Subscriber1: $error\n"; }, onDone: () { result += "Subscriber1: Stream closed!\n"; }); StreamSubscription<String> subscriber2 = stream.listen((String data) { result += "Subscriber2: $data\n"; }, onError: (error) { result += "Subscriber2: $error\n"; }, onDone: () { result += "Subscriber2: Stream closed!"; }); broadcastController.sink.add('UTU'); broadcastController.addError('Error...'); broadcastController.close(); }); }, child: const Text('Broadcast Stream')), // ElevatedButton const SizedBox(height: 20,), Text(result, style: const TextStyle(fontSize: 20,),), ```
12.	How to store data in SQLite databse using sqflite package? Write Dart code snippet and explain.
13.	How to read data from SQLite databse using sqflite package? Write Dart code snippet and explain.
14.	Analyze and explain following code snippet: ``` import 'package:path/path.dart'; import 'package:sqflite/sqflite.dart'; class MyDb { late Database db; Future open() async { // Get a location using getDatabasesPath var databasesPath = await getDatabasesPath(); String path = join(databasesPath, 'demo.db'); //join is from path package print(path); //output /data/user/0/com.example.sqflight_student/databases/demo.db db = await openDatabase(path, version: 1, onCreate: (Database db, int version) async { // When creating the db, create the table await db.execute(''' CREATE TABLE IF NOT EXISTS students(id primary key, name varchar(255) not null, roll_no int not null, address varchar(255) not null); '''); //table students will be created if there is no table 'students' print("Table Created"); }); ```
15.	Write necessary Dart code and explain rawInsert() and rawDelete() with respect to sqflite.
16.	Analyze and explain following code snippet:

```dart
import 'package:flutter/material.dart';
import 'db.dart';
class EditStudent extends StatefulWidget {
  final int rollno;
  const EditStudent({super.key, required this.rollno});

  @override
  State<StatefulWidget> createState() {
    return _EditStudent();
  }
}
class _EditStudent extends State<EditStudent> {
  TextEditingController name = TextEditingController();
  TextEditingController rollno = TextEditingController();
  TextEditingController address = TextEditingController();
  MyDb mydb = MyDb();
  @override
  void initState() {
    mydb.open();
    Future.delayed(const Duration(milliseconds: 500), () async {
      var data = await mydb.getStudent(widget.rollno);
      if (data != null) {
        name.text = data["name"];
        rollno.text = data["roll_no"].toString();
        address.text = data["address"];
        setState(() {});
      } else {
        print("Student NOT found with roll no: " + widget.rollno.toString());
```

17. What is Moor? Explain any four advantages of Moor.

18. Analyze and explain following code snippet (with respect to drift):

```dart
import 'dart:io';
import 'package:drift/drift.dart';
import 'package:drift/native.dart';
import 'package:path/path.dart' as p;
import 'package:sqflite/sqflite.dart' show getDatabasesPath;

part 'moor_database.g.dart';

class Tasks extends Table {
  IntColumn get id => integer().autoIncrement()();
  TextColumn get name => text().withLength(min: 1, max: 50)();
  DateTimeColumn get dueDate => dateTime().nullable()();
  BoolColumn get completed => boolean().withDefault(const Constant(false))();
}

@DriftDatabase(tables: [Tasks], daos: [TaskDao])
class AppDatabase extends _$AppDatabase {
  AppDatabase()
      : super(_openConnection());
  @override
  int get schemaVersion => 2;
}
```

19. Analyze and explain following code snippet (with respect to drift):

```dart
LazyDatabase _openConnection() {
  return LazyDatabase(() async {
    final dbFolder = await getDatabasesPath();
    final file = File(p.join(dbFolder, 'db.sqlite'));
    return NativeDatabase(file);
  }); // LazyDatabase
}

@DriftAccessor(
  tables: [Tasks],
  queries: {
    'completedTasksGenerated':
        'SELECT * FROM tasks WHERE completed = 1 ORDER BY due_date DESC, name;'
  },
)
```

20.	Step-by-step explain how to set app icon and launch screen in flutter app for deploying on Android platform.
21.	Step-by-step explain how to set app icon and launch screen in flutter app for deploying on iOS platform.
22.	
23.	
24.	